

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Технология машиностроения»

# СОВРЕМЕННЫЕ ТЕХНОЛОГИЧЕСКОЕ ОБОРУДОВАНИЕ И ОСНАСТКА ДЛЯ ИЗГОТОВЛЕНИЯ МАШИН, ТЕНДЕНЦИИ РАЗВИТИЯ

*Методические рекомендации к практическим занятиям  
для студентов специальности  
1-36 80 02 «Инновационные технологии в машиностроении»  
дневной и заочной форм обучения*



Могилев 2021

УДК 621.9  
ББК 34.4  
С56

Рекомендовано к опубликованию  
учебно-методическим управлением  
Белорусско-Российского университета

Одобрено кафедрой «Технология машиностроения» «31» марта 2021 г.,  
протокол № 9

Составитель канд. техн. наук, доц. А. В. Капитонов

Рецензент И. В. Лесковец

Методические рекомендации предназначены к практическим занятиям по дисциплине «Современные технологическое оборудование и оснастка для изготовления машин, тенденции развития» для студентов специальности 1-36 80 02 «Инновационные технологии в машиностроении». Даны указания для выполнения практических работ, целью которых является изучение и использование учебно-исследовательского комплекса на базе промышленного робота Kawasaki. Изложена последовательность управления и программирования робота. Сформулированы вопросы и задания.

Учебно-методическое издание

СОВРЕМЕННЫЕ ТЕХНОЛОГИЧЕСКОЕ ОБОРУДОВАНИЕ  
И ОСНАСТКА ДЛЯ ИЗГОТОВЛЕНИЯ МАШИН,  
ТЕНДЕНЦИИ РАЗВИТИЯ

Ответственный за выпуск	В. М. Шеменков
Корректор	А. А. Подошевка
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.  
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 21 экз. Заказ №

Издатель и полиграфическое исполнение:  
Межгосударственное образовательное учреждение высшего образования  
«Белорусско-Российский университет».  
Свидетельство о государственной регистрации издателя,  
изготовителя, распространителя печатных изданий  
№ 1/156 от 07.03.2019.  
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский  
университет, 2021

## Содержание

1 Общие положения.....	4
2 Практическое занятие № 1. Изучение структуры и компонентов учебного роботизированного комплекса.....	4
3 Практическое занятие № 2. Изучение методов ручного управления роботом.....	17
4 Практическое занятие № 3. Блочное программирование робота.....	22
5 Практическое занятие № 4. Изучение языка программирования AS робота Kawasaki. Разработка программ.....	27
6 Практическое занятие № 5. Изучение функций ввода команд программ с помощью редактора KRterm. Отладка и выполнение программ роботом.....	38
Список литературы.....	47

## **1 Общие положения**

### ***1.1 Цель практических занятий***

Целью практических занятий является изучение структуры и основных компонентов учебно-исследовательского комплекса на базе промышленного робота Kawasaki, приобретение навыков студентов в управлении и программировании промышленного робота.

### ***1.2 Порядок проведения занятий***

На практических занятиях студенты под руководством преподавателя изучают компоненты комплекса Kawasaki, обучаются управлению и программированию манипулятора Kawasaki FS003N, выполняют индивидуальные задания. В конце занятия студенты отвечают на вопросы, сформулированные в методических указаниях в устной или письменной форме.

## **2 Практическое занятие № 1. Изучение структуры и компонентов учебного роботизированного комплекса**

### ***2.1 Схема учебно-исследовательского комплекса на базе промышленного робота Kawasaki***

Схема учебно-исследовательского комплекса на базе промышленного робота Kawasaki представлена на рисунке 2.1

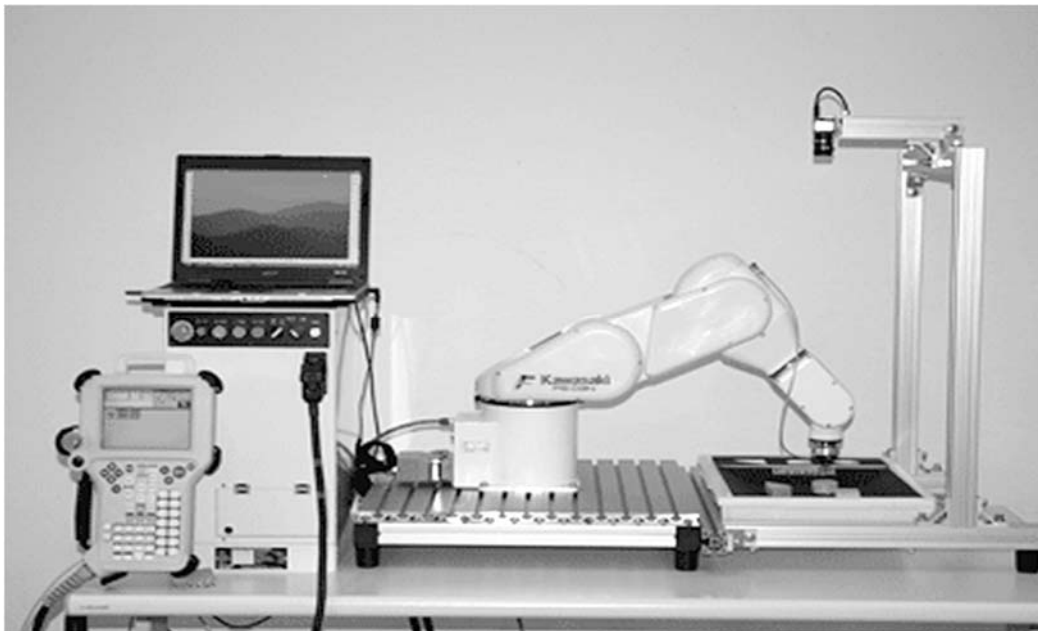


Рисунок 2.1 – Схема учебно-исследовательского комплекса

В состав комплекса входит: манипулятор Kawasaki FS003N; контроллер Kawasaki FD70; пульт оператора; вакуумный схват (установлен на манипулятор); монтажная панель; программа компьютерной симуляции Kawasaki PC ROSET.

Дополнительное оборудование – система технического зрения: видеокамера 2D; программное обеспечение обработки изображения RFA-Vision v4.1; ноутбук.

## 2.2 Описание контроллера

### 2.2.1 Панель контроллера.

Внешняя передняя панель контроллера или системы управления (СУ) робота (рисунки 2.2 и 2.3) осуществляет функции, описанные в таблице 2.1.

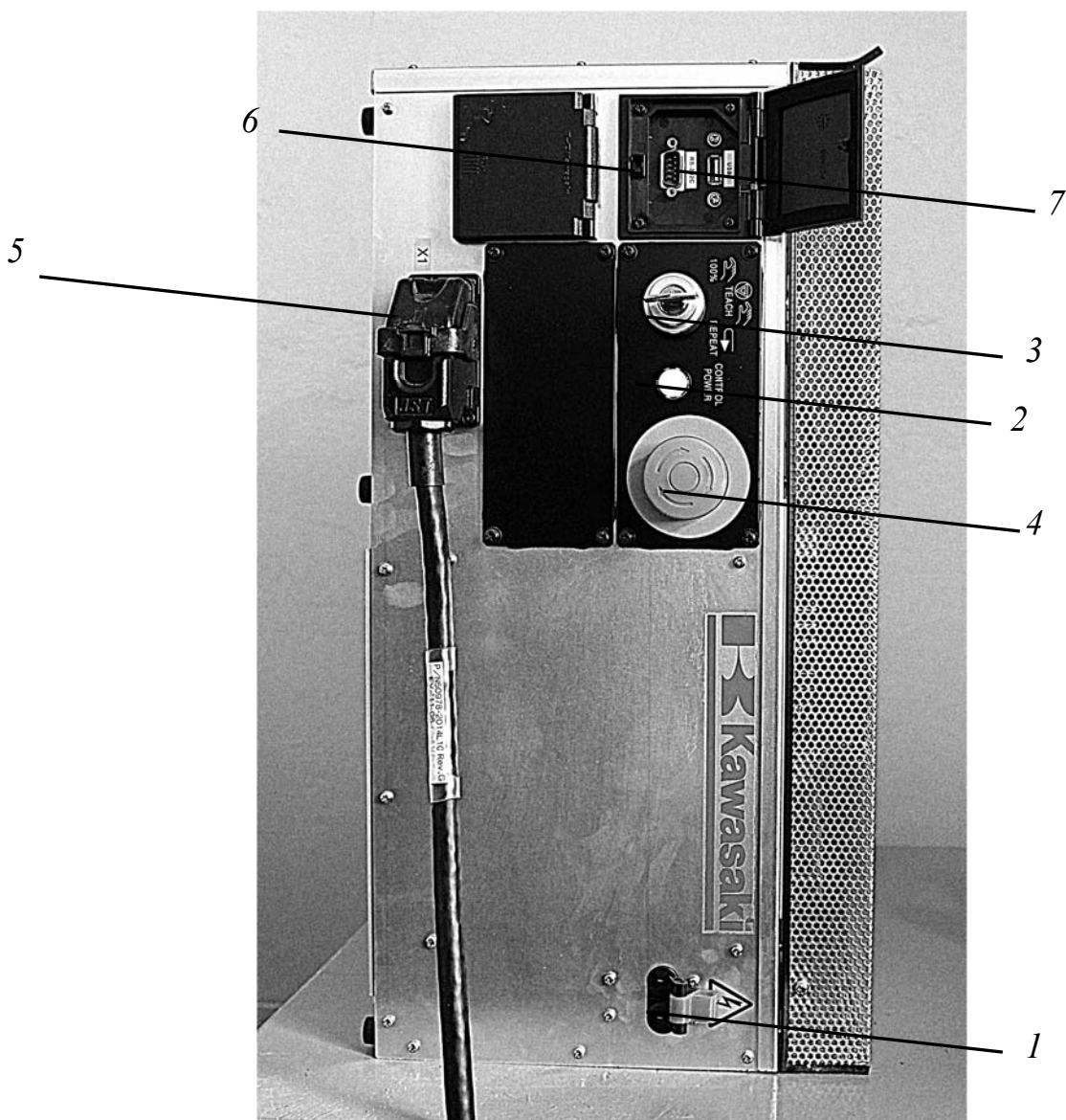


Рисунок 2.2 – Панель контроллера (СУ)

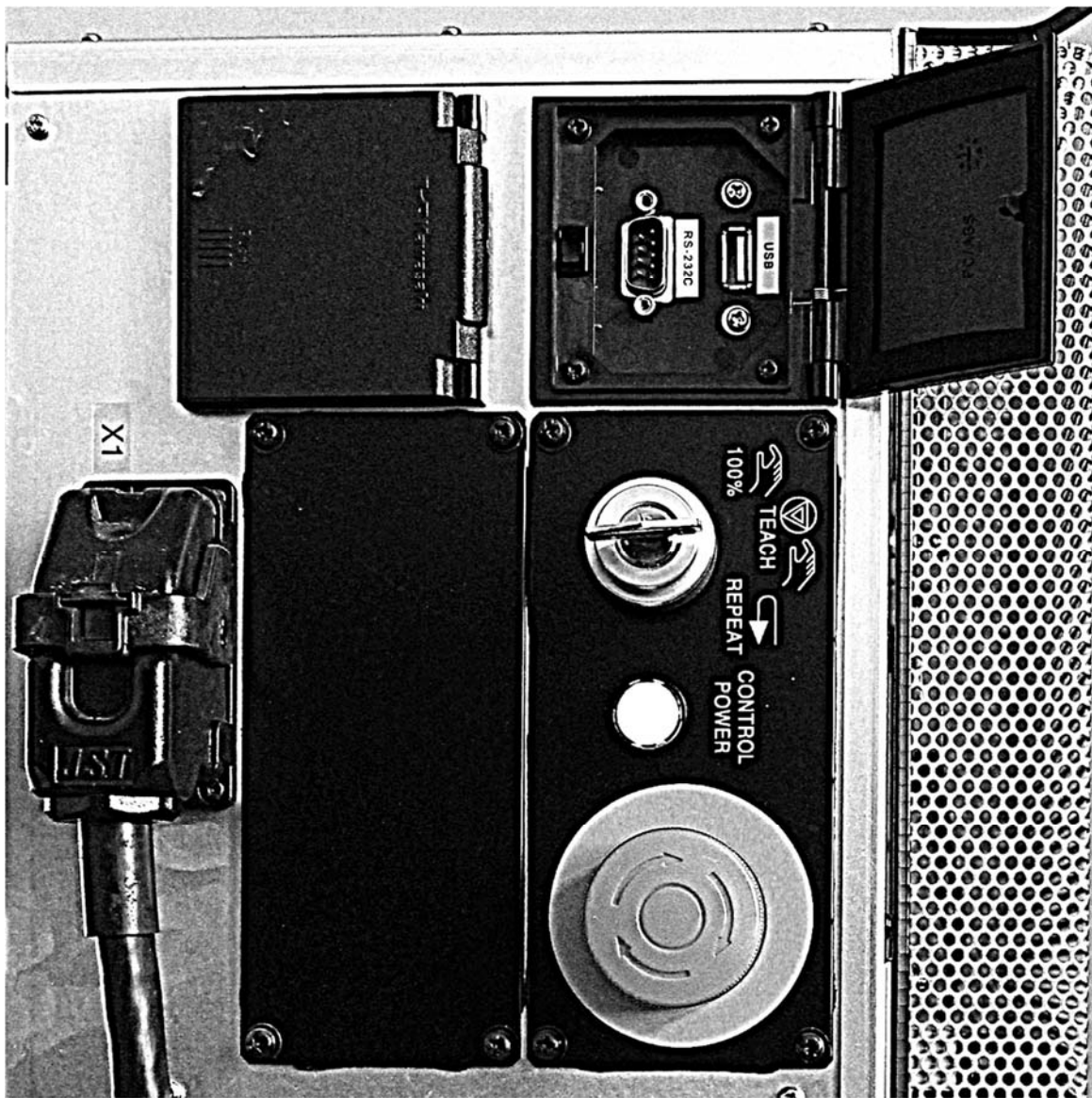


Рисунок 2.3 – Панель контроллера (верхняя часть)

Таблица 2.1 – Переключатели и лампочки на передней панели контроллера. Внешние входы/выходы на передней панели контроллера

Номер позиции на рисунке 2.2	Переключатель и лампочки	Назначение
1	2	3
1	Переключатель	Включает и выключает общее силовое питание
2	Индикатор источника силового питания	Загорается при включении общего силового питания
3	TEACH/REPEAT-переключатель	Переключает режимы «обучение» или «автоматический режим». Обеспечивает выбор режимов управления (режим обучения, когда переключатель находится в горизонтальном положении TEACH; режим выполнения программ, когда переключатель повернут по часовой стрелке и находится в положении REPEAT)

Окончание таблицы 2.1

1	2	3
4	Кнопка аварийного останова	Обрывает силовое питание привода и останавливает робот, когда эта кнопка нажата при непредвиденной ситуации. Силовое питание не отключается
5	Подключение первичного питания	Подключение кабеля первичного питания в контроллер
6	COM-порт	Подключение вспомогательных устройств и технологического оборудования
7	USB-порт	Подключение вспомогательных устройств

### 2.2.2 Включение силового питания контроллера.

Нужно переместить вправо переключатель силового питания (номер позиции 1 в таблице 2.1), расположенный в нижней правой части панели контроллера.

### 2.2.3 Включения и выключения силового питания привода робота.

Для включения силового питания двигателя робота нужно нажать клавиши A + Motor ON на пульте ручного управления. Затем нужно нажать клавиши A + RUN для разрешения движения. При этом окна MOTOR и RUN будут включены на экране пульта управления. Выключение силового питания привода осуществляется в обратном порядке: A + Motor ON, затем A + RUN или HOLD – останавливает движение.

### 2.2.4 Методы для останова робота.

Чтобы остановить робот различают методы останова для режима обучения, автоматического режима и аварийного останова.

#### Для режима обучения.

1 Отпустите переключатель TRIGGER на пульте ручного управления.

2 Убедитесь, что робот полностью остановился, и нажмите клавишу HOLD на пульте ручного управления. В правом верхнем углу экрана пульта управления появится окно HOLD вместо окна RUN.

#### Для автоматического режима.

Чтобы остановить движение робота при выполнении программы в автоматическом режиме, нужно нажать клавишу HOLD на пульте ручного управления.

Чтобы закончить выполнения шага по программе и затем остановить движение робота, нужно выбрать функцию Step Once нажатием в окне экрана пульта управления.

Чтобы закончить выполнения программы и затем остановить движение робота, нужно выбрать функцию Repeat Once нажатием в окне экрана пульта управления.

#### При аварийном останове.

Когда робот работает неправильно, существует возможная опасность столкновения робота с предметами получения травмы. В случае опасности нужно немедленно нажать любую кнопку аварийного останова на пульте ручного

управления или на операционной панели контроллера, чтобы отключить силовое питание привода.

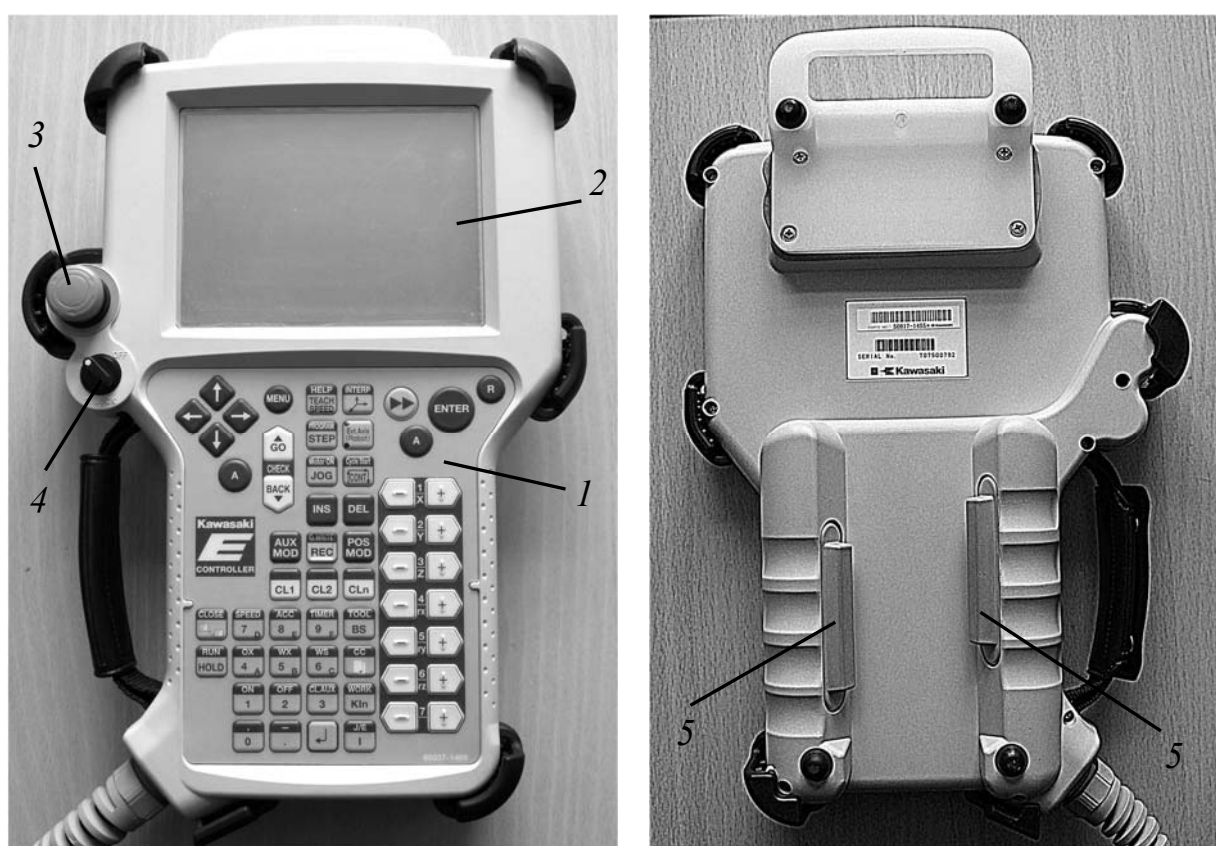
## 2.3 Пульт ручного управления роботом

### 2.3.1 Описание пульта ручного управления.

На рисунке 2.4 показан вид и краткое описание пульта ручного управления. Пульт ручного управления обеспечен аппаратными клавишами и переключателями, которые являются необходимыми для ручного управления роботом и данными редактирования, и экраном для того, чтобы отображать различные виды данных. Рисунок 2.5 показывает расположение аппаратных клавиш.

а)

б)



а – со стороны дисплея; б – задняя сторона пульта, со стороны триггера

Рисунок 2.4 – Вид пульта ручного управления

На рисунке 2.4 обозначено следующее: 1 – панель с клавишами; 2 – экран; 3 – кнопка аварийного останова; 4 – переключатель Teach Lock; 5 – переключатель TRIGGER.





Рисунок 2.5 – Расположение аппаратных клавиш на пульте управления

### 2.3.2 Аппаратные клавиши на пульте ручного управления.

Назначение основных аппаратных клавиш на пульте ручного управления и переключателей приведено в таблице 2.2.

Таблица 2.2 – Назначение основных аппаратных клавиш на пульте ручного управления и переключателей

Тип клавиши, переключателя	Назначение клавиши, переключателя
1	2
MENU	Отображает пошаговое меню выбранной области дисплея. Нажимая A + MENU активизируем пошаговое меню выбранной области дисплея
Стрелки вверх-вниз, влево-вправо	Передвигает курсор в вертикальном и горизонтальном направлениях между шагами, пунктами, строками. Одновременное нажатие «A + стрелка вверх или вниз» переводит курсор на предыдущий или последующий шаг в режиме обучения или редактирования

## Окончание таблицы 2.2

1	2
ENTER	Подтверждает выбор функции или пункта. Завершает ввод данных в активизированную строку выбранной области
R	Отмена последней операции. Закрывает пошаговое меню. Возврат в первоначальное состояние
A	«А» используется при выборе кнопок синего цвета (одновременное нажатие А +)
GO	Осуществляет пошаговое выполнение программных инструкций, переходя с выбранного шага к последующему
BACK	Осуществляет пошаговое выполнение программных инструкций, переходя с выбранного шага к предыдущему в проверочном режиме
TACH SPEED	Изменяет скорость движения манипулятора в режиме обучения
Оси координат	Задаёт систему координат: базовую (декартовую); относительную (движение по осям, связанным со схватом); угловую (движение отдельных суставов)
<u>PROGRAM</u> STEP	Нажимая А + PROGRAM, активизируем меню выбора программ. Нажимая STEP активизируем меню пошаговой области
Ext. Axis (Robot)	Выбирает дополнительную ось (JT7) или еще один робот, связанный с данным роботом
JOG	Медленно перемещает робот в режиме обучения или проверочном (контрольном режиме). Эффективно только при нажатой кнопке
CONT	Включает непрерывное или пошаговое выполнение программы
INS	Включает режим вставки программных шагов
DEL	Включает режим удаления программных шагов
AUX MOD	Включает режим редактирования дополнительной информации
POS MOD	Включает режим редактирования позиций
REC	Добавляет новый шаг следом за текущим шагом. Нажимая А+REC, записываем новый шаг в режиме обучения
CL1	Переключает зажим 1 в положение ON/OFF
CL2	Переключает зажим 2 в положение ON/OFF
CLn	Переключает n-й зажим в положение ON/OFF
Клавиши «+» и «←»	Двигает каждый сустав от JT1 до JT6
ON	Нажимая А + ON, устанавливаем выбранный зажим в положение ON
OFF	Нажимая А + OFF, устанавливаем выбранный зажим в положение OFF
CLAUX	Нажимая А + CLAUX в режиме обучения, вызываем дополнительные пункты для зажимов

### 2.3.3 Экран пульта ручного управления.

Пульт ручного управления обеспечен операционным дисплеем. Он состоит из трех областей А, В, С, как показано на рисунке 2.6.

#### Область экрана А.

Эта область состоит из пяти экранов. Их функции приведены ниже.

#### 1 Функции в *Program Comment*-области.

Эта область показывает зарегистрированное имя программы и его комментарии. Пошаговое меню в этой области имеет шесть функций (рисунок 2.7). Чтобы вызвать пошаговое меню нажмите A + PROGRAM на пульте ручного управления или нажмите область **Program Comment** непосредственно на экране.

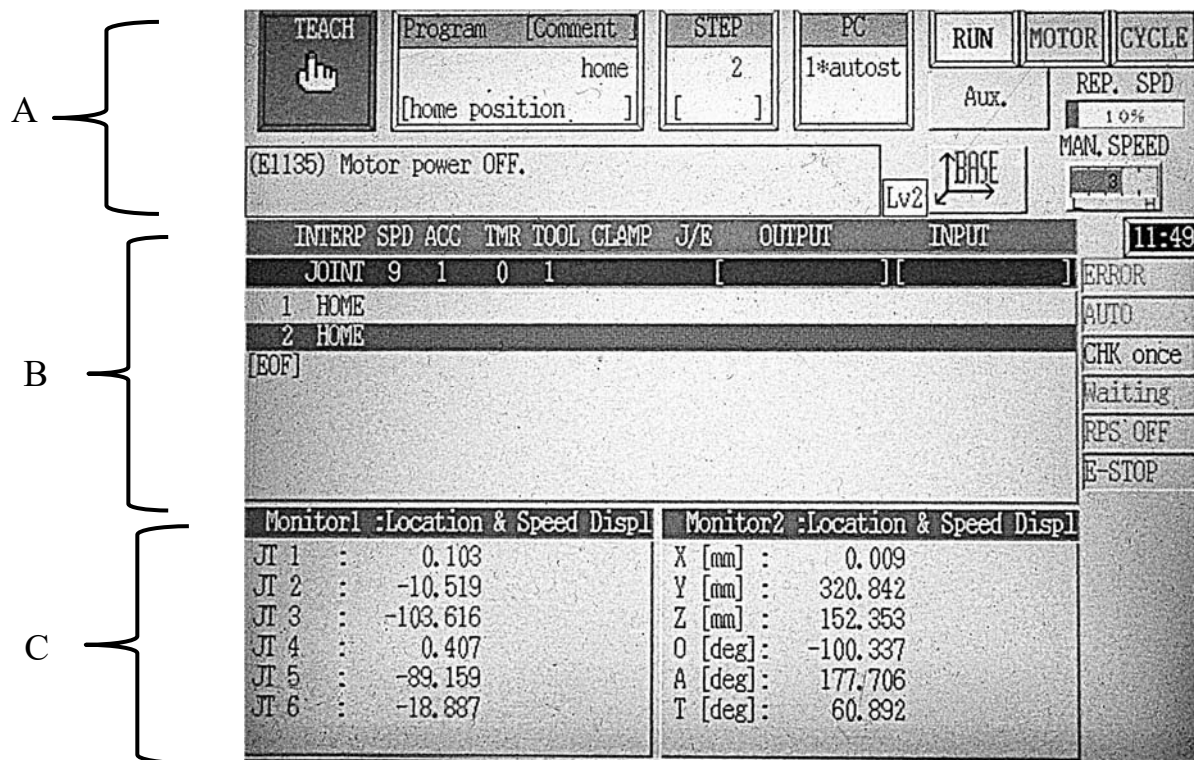


Рисунок 2.6 – Области дисплея

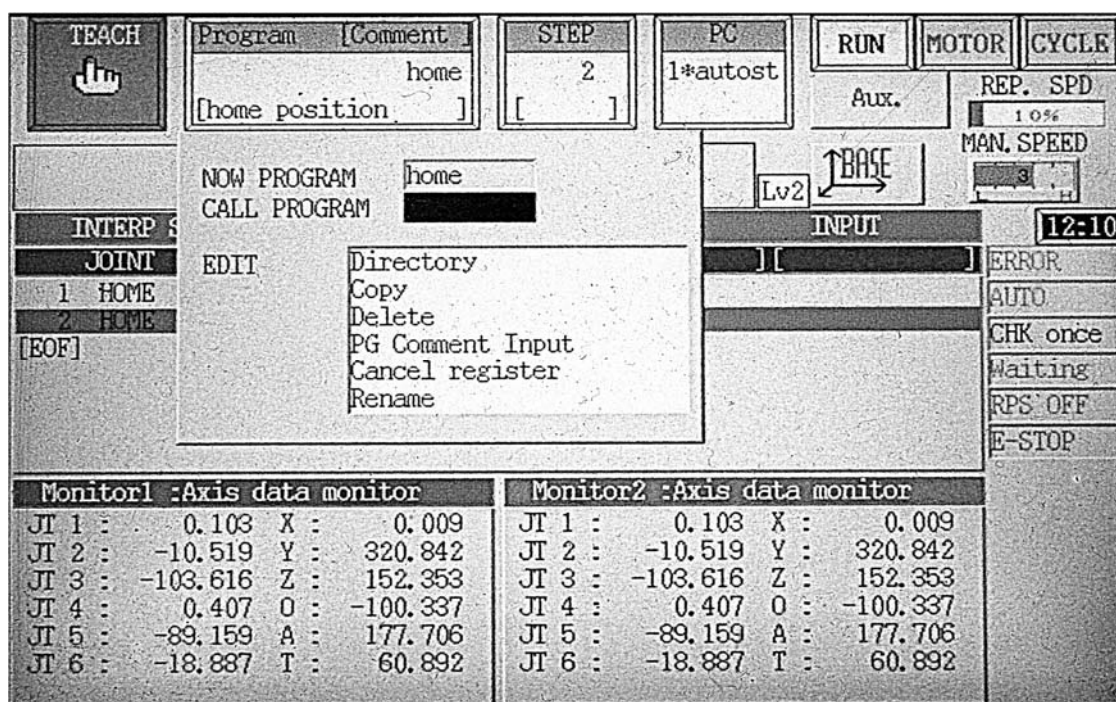


Рисунок 2.7 – Область PROGRAM COMMENT

В окно CALL PROGRAM можно ввести номер программы и выбрать программу, если имя программы состоит их чисел. Программы могут иметь имя до пяти символов.

Требуемая программа может быть выбрана из списка, если имя программы зарегистрировано. Нужно выбрать из пошагового меню функцию **Directory** (см. рисунок 2.7), после этого появится экран выбора программы (рисунок 2.8) и из списка нужно выбрать программу).

Directory		Program Name : 1	
PROGRAM NAME	STEP NUM	COMMENT	
1	0	( )	
autostart.pc	155	( )	
demo	31	( )	
get_picture	12	( )	
home	2	home position )	
kaw1	16	home position )	
kaw2	10	home position )	
kaw3	0	( )	
899	5	( )	
primer1	20	( )	
prog1	2	( )	
prog2	6	( )	
prog3	17	( )	
tcpcomm_close	17	( )	
tcpcomm_connect	50	( )	
tcpcomm_end	1	( )	
tcpcomm_recv	38	( )	
tcpcomm_send	41	( )	

Input                      Next Page

Рисунок 2.8 – Экран Directory выбора программы

Выбранная программа может быть удалена с помощью функции **Delete** (см. рисунок 2.7).

Комментарии могут быть введены в область комментариев, выбранной программы, с помощью функции **PG Comment Input**.

С помощью функции **Copy** можно скопировать содержание выбранной программы, изменив имя программы. Эту функцию можно использовать для создания новой программы.

С помощью функции **Cancel Register** программа может быть удалена из **Program Comment**-области.

## 2 Функции в **Step**-области.

Эта область показывает номер шага программы. Пошаговое меню в этой области имеет пять функций (рисунок 2.9). Чтобы показать пошаговое меню нужно нажать напрямую на экране пошаговую область.

На рисунке 2.9 в открытом пошаговом меню показан текущий шаг загруженной программы в поле NOW STEP – 16. Имя программы «kaw1» отображается в окне **Program Comment**. В поле вызова шага CALL STEP можно ввести

номер шага программы, нажать кнопку ENTER на пульте и этот шаг будет выбран (рисунок 2.10, выбран шаг 4).

Program	Comment	STEP	PC
kaw1	[home position. ]	16	1*autost

INTERP	SPD	ACC	TMR
JOINT 4	1	0	0
13 JOINT	4	1	0
14 JOINT	4	1	0
15 JOINT	4	1	0
16 JOINT	4	1	0

Monitor1 :Axis data monitor		Monitor2 :Axis data monitor			
JT 1 :	0.101 X :	0.002	JT 1 :	0.101 X :	0.002
JT 2 :	-10.519 Y :	320.842	JT 2 :	-10.519 Y :	320.842
JT 3 :	-103.616 Z :	152.353	JT 3 :	-103.616 Z :	152.353
JT 4 :	0.407 O :	-100.336	JT 4 :	0.407 O :	-100.336
JT 5 :	-89.159 A :	177.706	JT 5 :	-89.159 A :	177.706
JT 6 :	-18.887 T :	60.892	JT 6 :	-18.887 T :	60.892

Рисунок 2.9 – Step-область экрана

Program	Comment	STEP	PC
kaw1	[home position. ]	4	1*autost

Program held.No = 1

INTERP	SPD	ACC	TMR	TOOL	CLAMP	J/E	OUTPUT
JOINT 4	1	0	1				
1 JOINT	4	1	0	1			
2 JOINT	4	1	0	1			
3 JOINT	4	1	0	1			
4 JOINT	4	1	0	1			
5 JOINT	4	1	0	1			
6 JOINT	4	1	0	1			
7 JOINT	4	1	0	1			

Monitor1 :Axis data monitor		Monitor2 :Axis data monitor			
JT 1 :	0.811 X :	4.006	JT 1 :	0.811 X :	4.006
JT 2 :	-10.513 Y :	320.840	JT 2 :	-10.513 Y :	320.840
JT 3 :	-103.610 Z :	152.358	JT 3 :	-103.610 Z :	152.358
JT 4 :	0.385 O :	-100.502	JT 4 :	0.385 O :	-100.502
JT 5 :	-89.161 A :	177.708	JT 5 :	-89.161 A :	177.708
JT 6 :	-19.377 T :	60.944	JT 6 :	-19.377 T :	60.944

Рисунок 2.10 – Окна отработки программы в автоматическом режиме (пошаговая отработка)

Последний шаг в программе можно выбрать с помощью функции **Bottom** (рисунок 2.9, выбран последний шаг – 16).

### 3 Область системных сообщений.

Эта область показывает сообщения об ошибках и различные виды сообщений, которые возникают в течение работы робота. На рисунке 2.6 в окне системных сообщений указан код ошибки (E1135) и сообщение «Motor power OFF» (электропитание двигателя выключено).

### 4 Область состояния.

Эта область расположена справа от области РС вверху рабочего экрана на пульте ручного управления и включает следующее (см. рисунок 2.6):

- окно MAN. SPEED (manual speed) устанавливает скорость движения робота в режиме обучения (ручном режиме). Переключать скорость можно, нажав клавишу TEACH SPEED на пульте управления или касанием на экране в области окна;

- окно REP. SPD (Repeat speed) отображает скорость, когда робот перемещается в автоматическом режиме;

- три режима движения робота в режиме обучения:

- 1) BASE – базовая система координат;
- 2) TOOL – инструментальная система координат;
- 3) JOINT – режим суставного движения.

Нажимая клавишу с осями координат на пульте управления или соответствующее окно в области состояния, устанавливаем необходимую систему координат;

- окно MOTOR активно, когда силовое питание включено, и неактивно, когда силовое питание выключено;

- окно HOLD появляется, когда происходит останов робота, окно RUN – когда робот в состоянии выполнения;

- пошаговая отработка программы в режиме обучения. Нажав клавишу CONT на пульте управления, на экране справа в окне отобразится CHK ONCE. Выбирая шаг программы, нажав клавиши «A + стрелки вверх и вниз» на пульте управления, затем, нажав и удерживая триггер и клавишу GO, выполняем программу в пределах одного шага. Затем выбираем и выполняем второй шаг и т. д.;

- отработка программы в автоматическом режиме. Выбирая касанием на экране в окнах функции STEP CONT и REPEAT ONCE, выполняется загруженная программа только один раз. Выбирая функции в этих окнах STEP CONT и REPEAT CONT, выполняется загруженная программа непрерывно;

- пошаговая отработка программы в автоматическом режиме. Выбрав окно STEP ONCE на экране и, нажимая клавишу GO на пульте управления, выполняем программу пошагово (см. рисунок 2.10).

### Область экрана В.

Левая сторона этой области содержит пошаговое меню из пяти пунктов (рисунок 2.11).

Чтобы отобразить пошаговое меню, нужно нажать клавиши A + MENU на пульте управления или нажать окно области В непосредственно. Выбор Upsize

увеличивает и отображает содержание В- и С-областей.

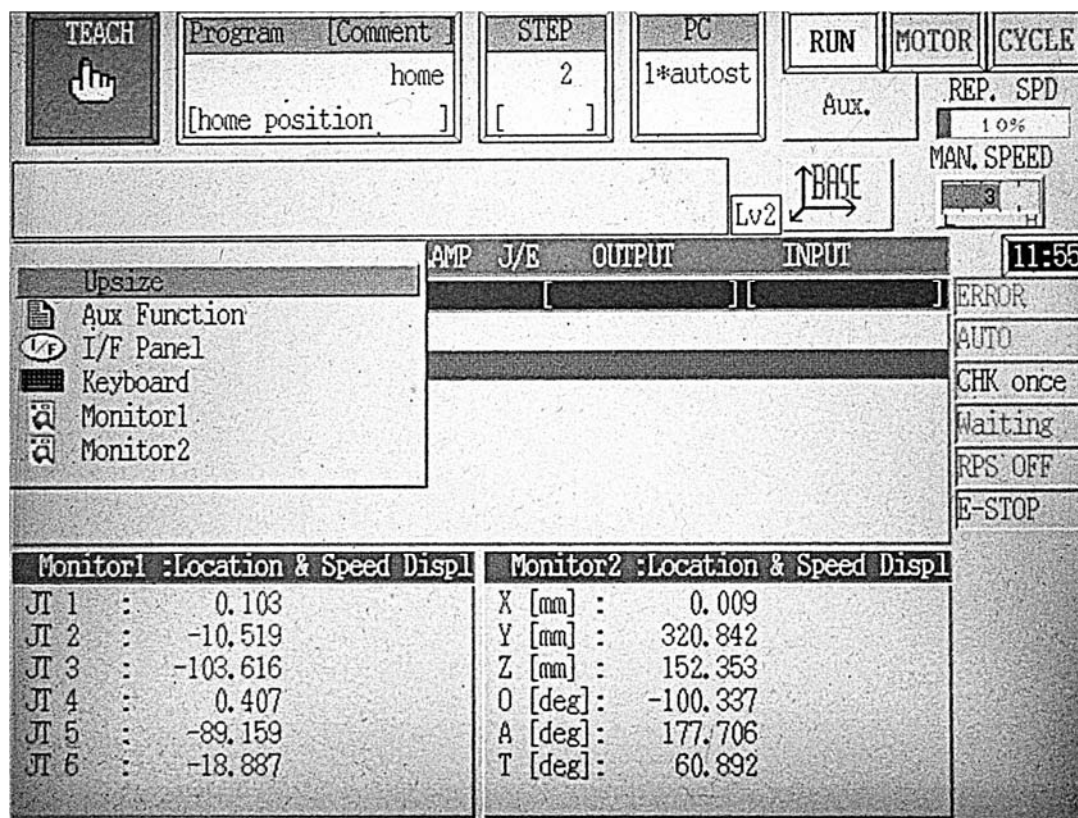


Рисунок 2.11 – Область экрана В

**Teach** – экран обучения. Используется для того, чтобы обучать или редактировать программу робота в блочном программировании (на рисунке 2.11 экран открыт).

**Aux Function** – экран вспомогательных функций. Используется, чтобы установить различные виды данных робота.

**I/F Panel** – экран интерфейсной панели. Этот экран обеспечивает выключатели и функции, необходимые для операций роботов и периферийного оборудования.

**Keyboard** – экран клавиатуры. Используется, чтобы ввести символы имени программы, комментариев и т. д.

**Monitor1** и **Monitor2** – экраны контроля открываются в области С.

### Область экрана С.

Эта область содержит два вида экранов, каждый из которых становится доступным из пошагового меню.

1 Экран контроля суставов. Этот экран отображает восемь видов информации, включая текущее положение робота, командные значения и т. д.

2 Экран контроля сигнала. Этот экран отображает предварительно установленные имена входов/выходов и контролирует состояние каждого сигнала.

## **2.4 Задания к практическому занятию**

1 Изучить панель контроллера Kawasaki FD70. Изучить функциональное назначение переключателей на панели, внешних входов/выходов.

2 Изучить назначение аппаратных клавиш на пульте ручного управления.

3 Изучить назначение и функционирование областей экрана А, В, С операционного дисплея пульта ручного управления.

4 Изучить методы включения/выключения контроллера, силового питания робота.

После изучения структуры и компонентов учебного роботизированного комплекса студенты под руководством преподавателя должны практически выполнить следующее:

- включить контроллер и силовое питание;
- на пульте управления выбрать и загрузить одну из программ из списка окна Directory, выбрать выборочно шаги программы;
- создать имя новой программы;
- установить различную скорость перемещения робота в ручном режиме;
- переключиться в каждый из трех режимов движения робота в режиме обучения;
- открыть меню в каждой области экрана А, В, С и изучить содержание функций в пошаговом меню и в открывающихся окнах;
- выключить силовое питание и контроллер.

### **Контрольные вопросы**

- 1 Что входит в состав учебно-исследовательского комплекса?
- 2 Какие функции выполняет переключатель TEACH/REPEAT?
- 3 Как выполняется включение и выключение силового питания привода робота?
- 4 Какие применяются методы для останова робота?
- 5 Из каких элементов состоит пульт ручного управления?
- 6 Из каких областей состоит операционный дисплей, их назначение?
- 7 Назовите три режима движения робота в режиме обучения.
- 8 Как выполняется пошаговая отработка программы в режиме обучения?
- 9 Как выполняется пошаговая отработка программы в автоматическом режиме?



## 3 Практическое занятие № 2. Изучение методов ручного управления роботом

### 3.1 Методы управления роботом вручную

#### 3.1.1 Имена осей.

Робот имеет шесть осей как показано на рисунке 3.1. Эти оси обозначаются JT1...JT6.

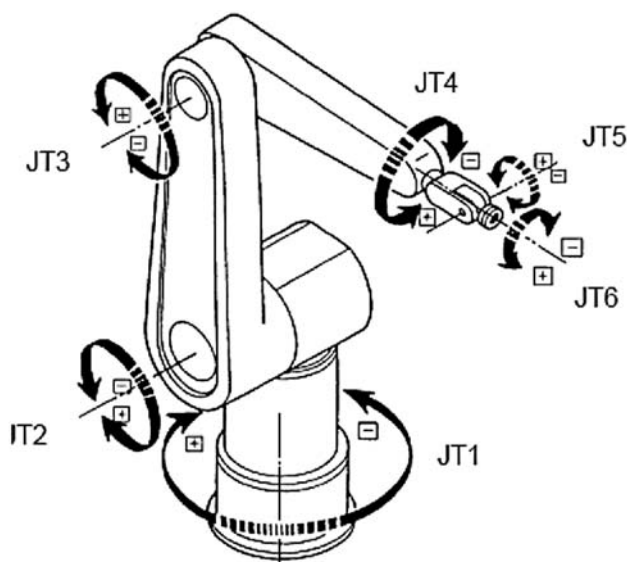


Рисунок 3.1 – Схема осей робота

#### 3.1.2 Задания к практическому занятию.

После изучения режимов ручного управления роботом студенты под руководством преподавателя выполняют операции по управлению каждой из шести осей.

Последовательность выполнения операций по управлению перемещений робота в ручном режиме.

1 Включите переключатель силового питания контроллера и убедитесь, что лампа силового питания включилась.

2 На панели контроллера поверните переключатель TEACH/REPEAT в положение TEACH (горизонтальное положение).

3 Включите в положение ON переключатель TEACH LOCK на пульте ручного управления.

4 Нажмите клавишу с координатными осями на пульте ручного управления или непосредственно в окне области состояния дисплея, чтобы установить режим операции: Joint, Base или Tool.

5 Установите скорость операции, нажимая клавишу TEACH SPEED на пульте или окно в области состояния на дисплее.

6 Когда шаги 1–5 закончены, включите силовое питание привода (A + Motor ON).

7 Поверните переключатель HOLD/RUN в позицию RUN.

8 Пока нажат переключатель TRIGGER на пульте ручного управления

можно перемещать робот, нажимая клавиши осей «+» и «-» от 1 до 6. Робот продолжит двигаться, пока клавиши нажаты.

9 При отпускании клавиш «+» и «-» или переключателя TRIGGER на пульте ручного управления робот останавливается.

После завершения ручного управления установите в положение OFF переключатель TEACH LOCK на пульте ручного управления.

### 3.1.3 Ручное управление седьмой осью.

Седьмая ось – дополнительная ось, которая добавляется и может быть продольной осью, сварочные клещи с серводвигателем и т. д. Ручное управление такое же, как при действии с шестью осями.

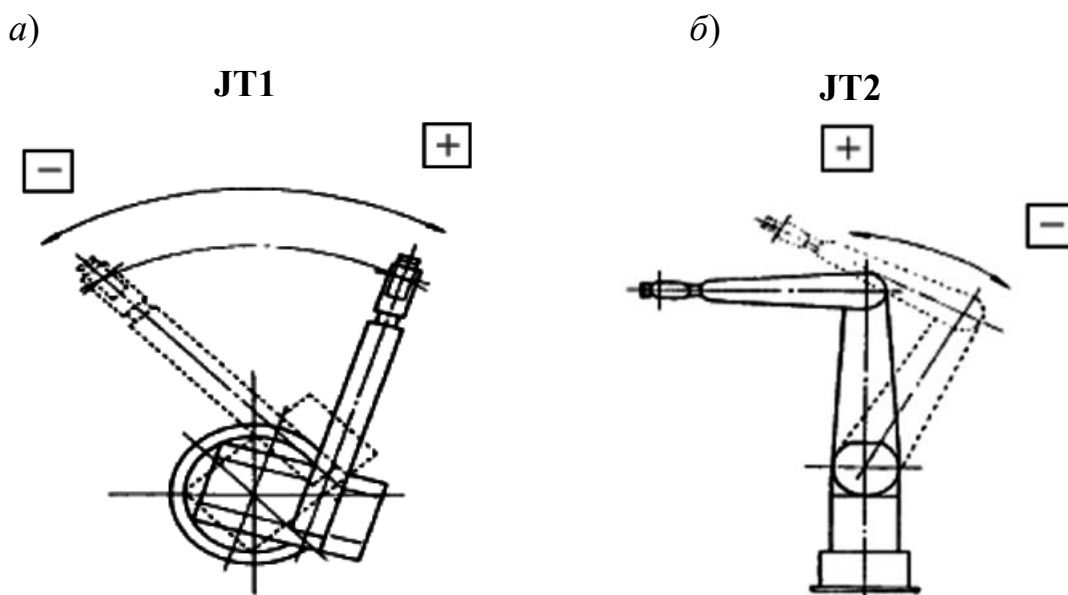
### 3.1.4 Ручное управление 8–18-ми осями.

Этот контроллер может управлять максимумом 18 осями. Ручное управление этими осями такое же, как для седьмой оси. Для того чтобы использовать дополнительные оси, или 8–14 или 15–18, нажимают EXT AXIS (Robot).

## 3.2 Режимы ручного управления робота

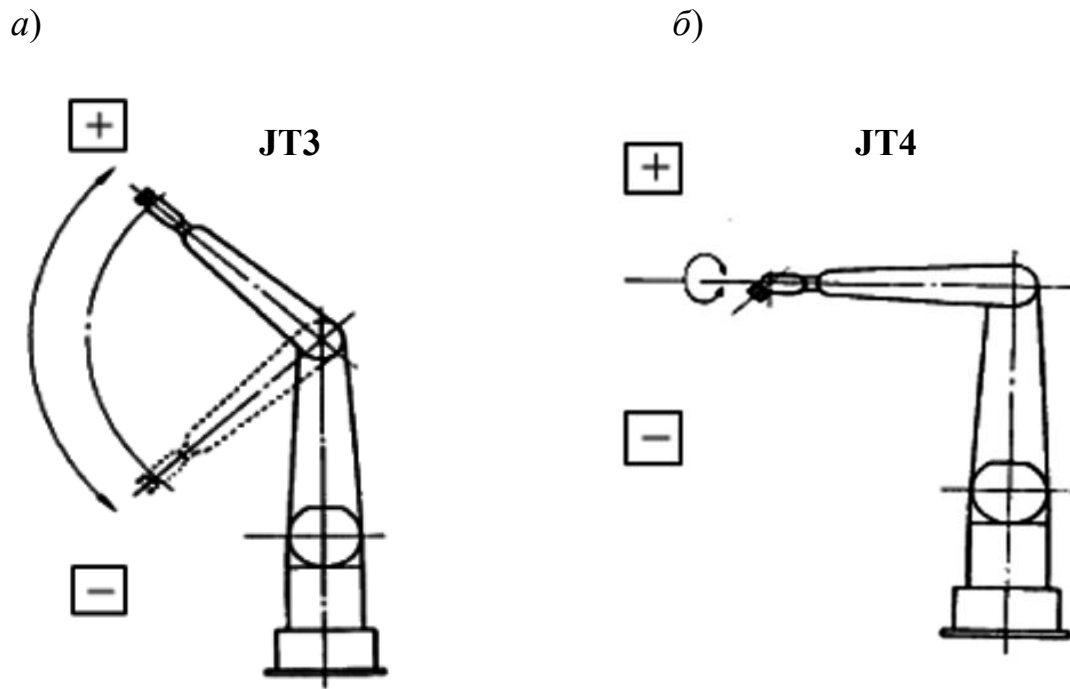
### 3.2.1 Режим суставных координат.

Нажав клавишу с координатными осями на пульте ручного управления или непосредственно нажав в окне области состояния дисплея, можно изменить изображение текущего режима на режим суставных координат Joint. Когда этот режим выбран, оси робота могут перемещаться индивидуально, как показано на рисунке 3.1. Когда нажимается несколько клавиш осей одновременно, оси робота могут перемещаться в комбинации (рисунки 3.2–3.4).



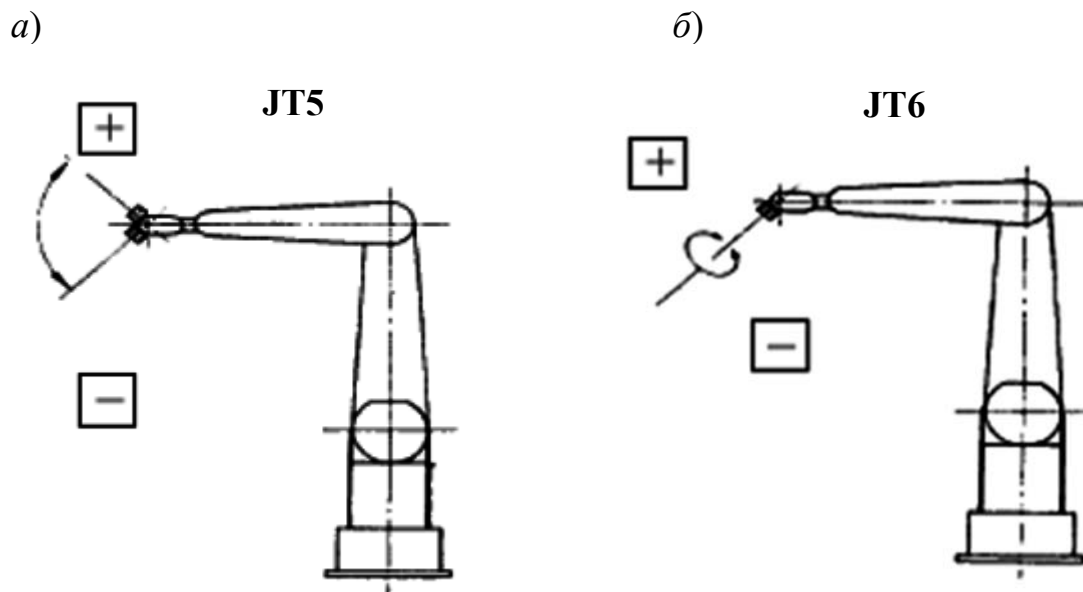
*а* – вращение руки влево и вправо вокруг оси JT1; *б* – движение назад и возврат руки вокруг оси JT2

Рисунок 3.2 – Движение суставов робота вокруг осей



*a* – движение руки вверх и вниз вокруг оси JT3; *б* – вращение запястья вокруг оси JT4

Рисунок 3.3 – Движение суставов робота вокруг осей



*a* – движение запястья вверх и вниз вокруг оси JT5; *б* – вращение лицевой части запястья вокруг оси JT6

Рисунок 3.4 – Движение суставов робота вокруг осей

### 3.2.2 Режим основных координат.

Изменим изображение режима на режим основных координат Base с помощью пульта ручного управления или нажатием области в окне дисплея. Когда этот режим выбран, оси робота могут перемещаться, базирясь на основной системе координат. Когда нажато несколько клавиш одновременно, оси робота

могут перемещаться в комбинации. В базовой системе координат робот может перемещаться параллельно осям и вращаться вокруг осей (рисунок 3.5).

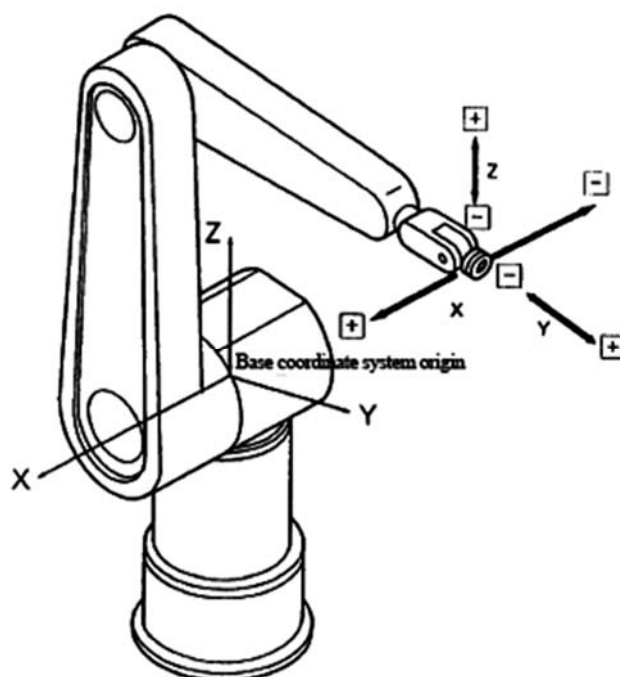


Рисунок 3.5 – Основная базовая и инструментальная системы координат

### 3.2.3 Режим инструментальных координат.

Изменим изображение режима на режим инструментальных координат Tool (рисунок 3.6) с помощью пульта ручного управления или нажатием области в окне дисплея. Когда этот режим выбран, оси робота могут перемещаться, базирясь на системе координат инструмента.

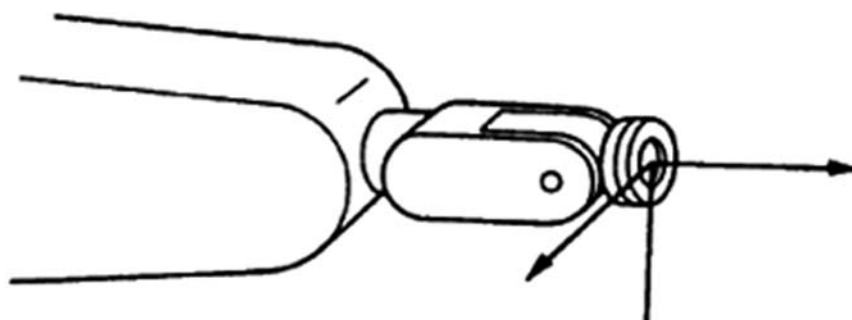


Рисунок 3.6 – Горизонтальное положение робота относительно сустава J3 в режиме инструментальных координат

Система координат инструмента использует набор координат, который определяет положение инструмента в пространстве. Установка инструментальной системы координат меняется каждый раз, когда позиция и положение робота меняется.

Значения инструментальной системы координат должны быть изменены, когда используется новый инструмент, имеющий другую форму.

На рисунке 3.7 показано как меняется положение системы координат в зависимости от положения в пространстве робота. При горизонтальном положении робот имеет инструментальную систему координат  $X_1, Y_1, Z_1$ . При перемещении робота вниз положение инструментальной системы координат изменится на  $X_2, Y_2, Z_2$ .

Когда положение запястья меняется, положение инструментальной системы координат также меняется.

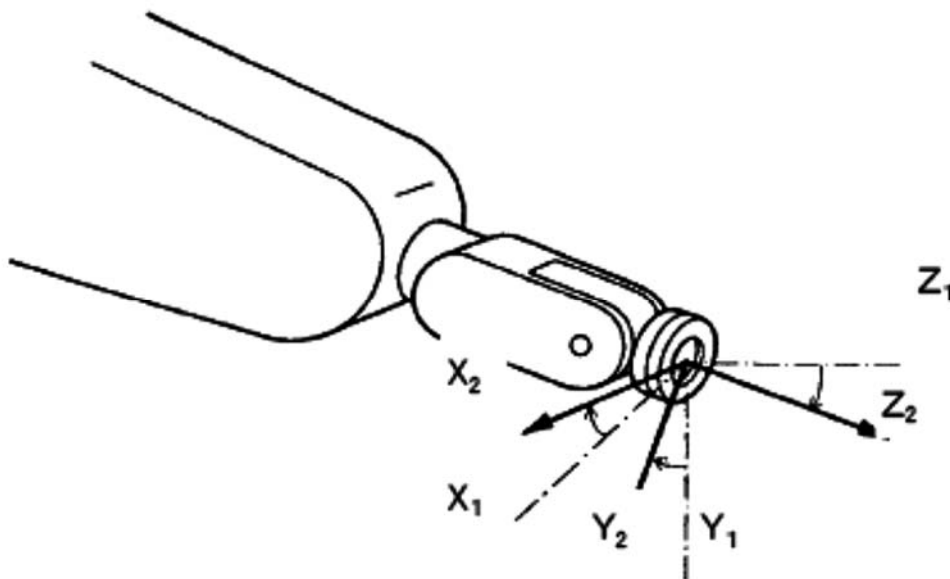


Рисунок 3.7 – Положение робота, перемещенного вниз относительно сустава JT3, в режиме инструментальных координат

### ***Контрольные вопросы***

- 1 Сколько осей имеет робот и как они обозначаются?
- 2 Как перемещается робот в режиме суставных координат?
- 3 Как перемещается робот в режиме основных координат?
- 4 Как перемещается робот в режиме инструментальных координат?
- 5 Сколько дополнительных осей может иметь робот?
- 6 Как направлены оси в базовой и инструментальной системах координат?

## 4 Практическое занятие № 3. Блочное программирование робота

Метод блочного программирования заключается в обучении робота и редактировании программы, используя пульт ручного управления.

### 4.1 Подготовка для обучения робота

#### 4.1.1 Установка переключателя TEACH LOCK.

В начале обучения нужно установить переключатель TEACH LOCK на пульте ручного управления в положение ON и переключатель на контроллере TEACH/REPEAT в положение TEACH.

Робот не будет двигаться, если переключатель на контроллере TEACH/REPEAT установлен в автоматический режим REPEAT. Кроме того, если переключатель на пульте ручного управления TEACH LOCK находится в состоянии OFF, робот не может перемещаться вручную в режиме обучения.

#### 4.1.2 Конфигурация экрана обучения.

При обучении программ в блочном программировании используется экран обучения.

Чтобы отобразить экран обучения, нужно активизировать В-область экрана и нажать клавиши пульта управления A + MENU или окно В-области, чтобы отобразить пошаговое меню, затем выбрать пункт TEACH из меню.

Конфигурация экрана обучения следующая (рисунок 4.1).

Главную (самую высокую) строку называют строкой заголовка, и пункты, необходимые для блочного программирования, отображаются в этой строке.

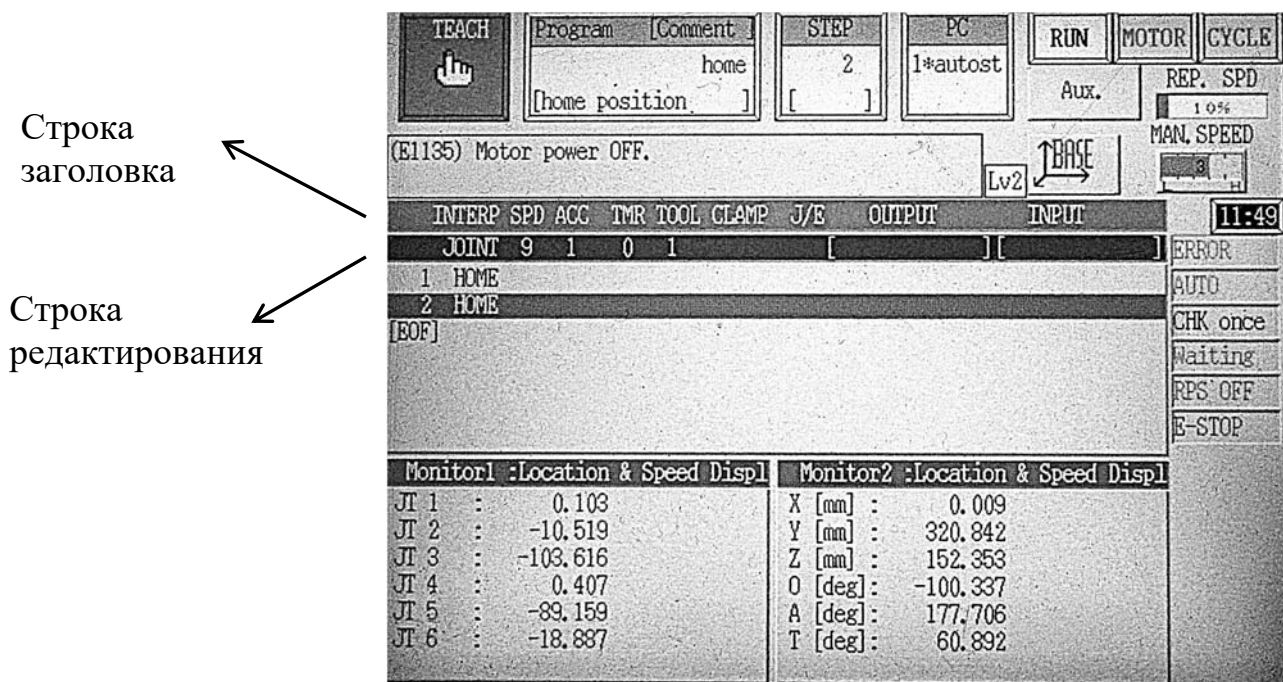


Рисунок 4.1 – Конфигурация экрана обучения

Строка редактирования расположена ниже строки заголовка. Эта строка используется, чтобы редактировать содержание каждого шага.

Для переключения из одного столбца (элемента) строки заголовка в другой нужно переместить курсор при помощи клавиш со стрелками «влево-вправо» пульта управления. Изменения значений вспомогательных данных (элементов) производятся с помощью клавиш со стрелками «вверх-вниз» на пульте управления или вводом чисел напрямую.

Содержание каждого элемента в строке заголовка показано в таблице 4.1.

Таблица 4.1 – Содержание элемента в строке заголовка экрана обучения

Название элемента (вспомогательных данных)	Содержание элемента
Interpolation	Определяет способ перемещения манипулятора к каждой обученной точке непрерывно. Для примера, выбор линейной интерполяции сделает траекторию между точками прямолинейной
Speed	Задаёт скорость движения в обученную точку
Accuracy	Определяет порядок, в котором робот приближается, и считает, что достиг точку
Timer	Определяет время ожидания в обученной точке
Tool	Определяет номер рабочего инструмента, смонтированного на работе
Clamp	Задаёт ON/OFF состояние схвата при взятии детали
J/E	Задаёт переключение программ при помощи внешнего сигнала
O/X	Задаёт выходной сигнал от робота к периферийному оборудованию
W/X	Задаёт входной сигнал к роботу от периферийного оборудования

## 4.2 Операции обучения робота

Обучение робота методом блочного программирования выполняется на экране обучения, используя пульт ручного управления.

### Пример обучения робота.

В качестве примера рассмотрим как обучить робот, который будет перемещаться по заданной траектории к каждой из четырех точек, как показано на рисунке 4.2.

Сначала нужно создать имя новой программы или загрузить имеющуюся программу, которая должна быть отредактирована.

Задание на обучение робота показано в таблице 4.2.

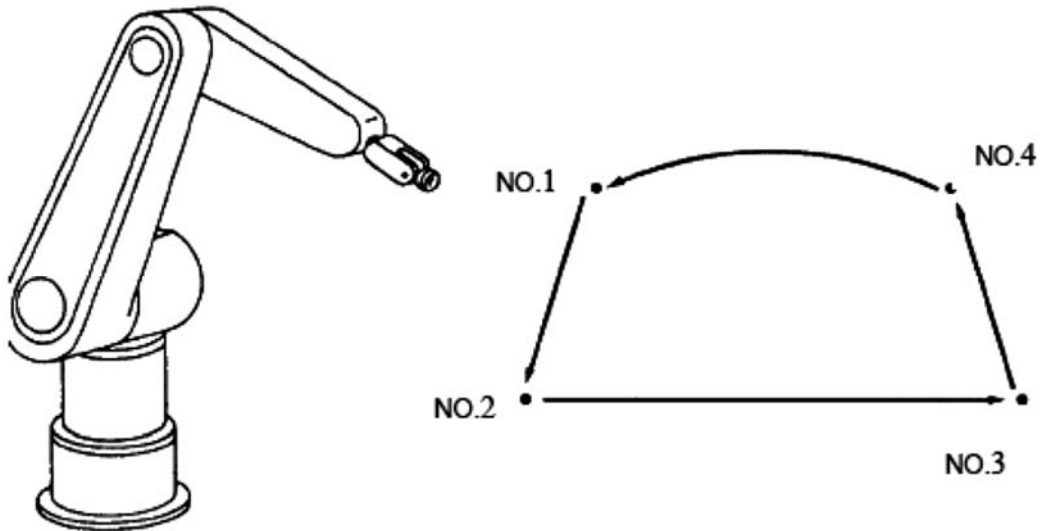


Рисунок 4.2 – Обучение робота по заданной траектории

Таблица 4.2 – Содержание процедуры обучения робота

Обучаемый шаг программы	Содержимое обучения
1	Перемещение в начальную точку робота N0.1
2	Движение из точки N0.1 в точку N0.2 с медленной скоростью по линейной траектории, с высокой точностью, с небольшой задержкой
3	Движение от точки N0.2 в точку N0.3 с медленной скоростью по линейной траектории со сменой инструмента
4	Движение из точки N0.3 в точку N0.4 с более высокой скоростью по прямолинейной траектории
5	Движение из точки N0.4 в точку N0.1 с более высокой скоростью по траектории Joint

### Последовательность выполнения задания.

1 Перемещаем робот в точку N0.1 для обучения шага 1 при помощи клавиш «+» и «-», удерживая переключатель TRIGGER. Нажимаем клавишу REC на клавиатуре пульта ручного управления (ПРУ) для того, чтобы обучить позицию. Скорость, точность перемещения и вид траектории не устанавливаем, они остаются текущими.

2 Перемещаем робот из точки N0.1 в точку N0.2 для обучения шага 2 при помощи клавиш «+» и «-», удерживая переключатель TRIGGER. Устанавливаем линейную интерполяцию, скорость 3, точность 7, таймер 1 следующим образом.

Для установки линейной интерполяции перемещаем курсор в заглавной строке на колонку INTERP (см. рисунок 4.1), нажимая клавиши (ПРУ) A + INTERP или при помощи клавиш со стрелками «вправо-влево». Нажимая клавишу (ПРУ) со стрелкой вверх или нажимая на экране область INTERP, изменяем режимы, установленные в строке редактирования на режим LINEAR.



Для установки скорости движения робота перемещаем курсор в заглавной строке на колонку SPD, нажимая кнопки A + SPEED или при помощи клавиш со стрелками «вправо-влево». Устанавливаем скорость 3 с помощью клавиш со стрелками «вверх-вниз» или нажимая на экране область SPD.

Для установки точности перемещения робота перемещаем курсор в заглавной строке на колонку Acc (точность), нажимая клавиши A + ACC или при помощи клавиш со стрелками «вправо-влево». Устанавливаем точность 7 с помощью клавиш со стрелками «вверх-вниз» или нажимая на экране область ACC.

Для установки таймера перемещаем курсор в заглавной строке на колонку TMR, нажимая клавиши A + TIMER или при помощи клавиш со стрелками «вправо-влево». Устанавливаем таймер 1 с помощью клавиш со стрелками «вверх-вниз» или нажимая на экране область TMR. Нажимаем клавишу REC на клавиатуре ПРУ для того, чтобы обучить позицию.

3 Перемещаем робот из точки N0.2 в точку N0.3 для обучения шага 3 при помощи клавиш «+» и «-», удерживая переключатель TRIGGER. Не меняем линейную интерполяцию, скорость 3 и точность 7. Таймер не устанавливаем.

Меняем инструмент, устанавливаем инструмент 2. Для установки инструментальной системы координат перемещаем курсор в заглавной строке на колонку TOOL, как в предыдущем шаге. Изменяем инструментальную систему координат в строке редактирования, нажимая клавиши A + TOOL или при помощи клавиш со стрелками «вправо-влево». Устанавливаем номер инструмента 2. Нажимаем клавишу REC на клавиатуре ПРУ для того чтобы обучить позицию.

4 Перемещаем робот в точку N0.4 для обучения шага 4. Устанавливаем линейную интерполяцию, скорость 6, точность 3, как в предыдущих шагах. Нажимаем клавишу REC на клавиатуре ПРУ для того, чтобы обучить позицию.

5 Перемещаем робот в точку N0.5 для обучения шага 5. Устанавливаем по суставную интерполяцию JOINT в колонке INTERP, скорость и точность не изменяем. Режим интерполяции устанавливаем так же, как во втором шаге. Нажимаем клавишу REC на клавиатуре ПРУ для того, чтобы обучить позицию.

Операции обучения для данной программы завершены.

Каждый шаг программы должен заканчиваться нажатием клавиши REC для записи в память робота.

### ***4.3 Проверка работы программы***

Для того чтобы подтвердить воспроизведение обученной программы, используются клавиши пульта управления GO и VACK в режиме проверки.

1 Выбираем программу, которая должна быть проверена.

2 Можно начать проверку с первого шага или с любого выбранного шага, для этого устанавливаем шаг программы.

3 Переключаемся в режим обучения и переключаем TEACH LOCK переключатель на пульте ручного управления в состояние ON. Устанавливаем TEACH/REPEAT-переключатель в положение TEACH.

Нажатие клавиши CONT на пульте управления переключает метод проверки, выполнение одного приращения шага или непрерывное выполнение. Метод проверки отображен в области состояния.

4 Устанавливаем скорость проверки.

5 После включения силового питания привода, нужно нажать клавишу RUN на пульте ручного управления и переключиться на окно RUN на дисплее, для того чтобы перемещать робот при помощи пульта ручного управления.

6 Нажмите переключатель TRIGGER на пульте ручного управления и нажмите клавишу GO, чтобы переместить робот к выбранному шагу.

7 Когда проверяется приращение в одном шаге ONCE, робот останавливается по завершению движения.

8 При непрерывной проверке шагов CONT, робот перемещается непрерывно в следующий шаг, в то время как GO нажато.

#### ***4.4 Редактирование данных программы***

Есть четыре основных процедуры для редактирования обученных данных программы.

1 Перезапись позиции.

2 Перезапись вспомогательных данных.

3 Вставка шагов.

4 Удаление шагов.

Перезапись позиции выполняется в следующей последовательности.

1 Устанавливаем курсор на шаг, который нуждается в редактировании, нажимая клавиши на пульте управления A + клавиши со стрелками «вверх-вниз». Движение к шагу видно по перемещению строки зеленого цвета.

2 Нажимаем клавиши на пульте управления A + POS/MOD, цвет строки шага меняется и в левой части строки редактирования появится изображение POS.M.

3 Подводим манипулятор к корректируемой позиции, нажимая клавиши на пульте управления «+» и «-».

4 Нажимаем клавишу REC, тем самым регистрируем новую позицию шага. Редактирование закрывается нажатием клавиши A + POS/MOD.

Перезапись вспомогательных данных выполняется с помощью клавиши пульта управления A + AUX/MOD. Вставка шагов выполняется с помощью клавиши A + INS. Удаление шагов выполняется с помощью клавиши A + DEL. Последовательность выполнения этих функций аналогична перезаписи позиций.

#### ***4.5 Запуск выбранной программы в автоматическом режиме***

Рассмотрим основные методы для запуска робота в автоматическом режиме, используя пульт ручного управления. Эти методы называют операциями повтора.

1 Загружаем программу, которую хотим выполнить.

2 Устанавливаем главный переключатель контроллера TEACH/REPEAT

в положение REPEAT.

3 Устанавливаем переключатель пульта ручного управления TEACH LOCK в положение OFF.

4 Устанавливаем необходимые состояния выполнения программы (непрерывно, выполнение только один раз, пошаговое выполнение).

5 Нажимаем клавиши A + Motor ON на пульте управления (включаем привод робота). Кнопка MOTOR на дисплее станет активной.

6 Нажимаем клавиши A + RUN, должна отобразиться на дисплее кнопка RUN (даем разрешение на выполнение программы).

7 Нажимаем CYCLE START на контроллере. Кнопка CYCLE на дисплее станет активной. Робот начинает выполнение программы.

Когда TEACH LOCK-переключатель на пульте в состоянии ON, автоматический режим невозможен.

Есть два способа для того, чтобы остановить робот во время автоматического режима: прерывание программы или завершение выполнения программы.

#### **Способы прерывания программы.**

1 Нажать клавишу HOLD на пульте ручного управления.

2 Когда робот заканчивает шаг движения полностью, установить переключатель пульта управления TEACH LOCK в положение ON, что отключит силовое питание привода.

3 Когда робот заканчивает шаг движения полностью, нажать любую аварийную кнопку для выключения силового питания привода.

Для завершения выполнения программы нужно установить состояние REPEAT CONT/ONCE в положение ONCE в окне дисплея.

### **4.6 Задания к практическому занятию**

1 Выполнить задание, представленное в таблице 4.2, следуя описанным инструкциям по обучению робота.

2 Выполнить индивидуальное задание по блочному программированию робота. Задание выдается преподавателем.

## **5 Практическое занятие № 4. Изучение языка программирования AS робота Kawasaki. Разработка программ**

### **5.1 Основы программирования робота на языке AS**

Роботы Kawasaki управляются системной программой, называемой AS. В AS-системе можно задавать команды или выполнять программы, используя AS-язык программирования. AS-система записана в энергонезависимой памяти контроллера, в оперативном запоминающем устройстве (ОЗУ). Когда управляющее питание включается, AS-система запускается и ожидает ввода команд для выполнения.

AS-язык программирования разделяется на два вида: мониторные команды и программные инструкции.

Мониторные команды используются для написания, редактирования, выполнения программ и единичных команд. Они вводятся после знака «>», появляющегося в начале строки, и выполняются после нажатия клавиши **Enter** немедленно. Некоторые мониторные команды применяются внутри программы как программные инструкции.

Программные инструкции используются для создания последовательности движения робота, для контроля и управления внешними сигналами в программах.

*Программа есть совокупность программных инструкций.*

В AS-системе возможны два способа задания координатной точки в трехмерном пространстве:

1) последовательностью угловых смещений каждой из осей суставов относительно выбранного начального положения (угловая координата);

2) последовательностью линейных и угловых смещений относительно начала базовой системы координат (декартова координата).

При запоминании точки в угловых координатах необходимо перед буквой поставить знак #.

*Пример записи угловой координаты:*

		JT1	JT2	JT3	JT4	JT5	JT6
#pose	=	0	33	-15	0	-40	30

Здесь JT1...JT6 – оси суставов (шарниров) робота.

*Пример записи декартовой координаты:*

		X	Y	Z	O	A	T
pose	=	0	1434	300	0	0	0

Координата задается значениями X, Y, Z для основных координат и положением O, A, T углов Эйлера. Имя декартовой координаты начинается с буквы.

Каждая строка (шаг) программного языка AS выражена в следующем формате:

**номер шага      метка      программные инструкции;      комментарии**

*Номер шага* назначается для каждой строки программы. Шаги нумеруются последовательно, начиная с 1 (при вводе с терминала автоматически нумеруются и перенумеровываются каждый раз, когда строки вставляются или удаляются).

*Метка* появляется при записи или редактировании программы с терминала в виде значка «?». При записи программы вне терминала метка не вводится.

*Программные инструкции* – это команды для перемещения робота и другие внешние команды.

Точка с запятой (;) указывает, что вся информация направо от точки с запятой (;) – это *комментарии*.

*Глобальные переменные* – это переменные, которые определены и сохраняются с этим значением в памяти. Они могут использоваться в любой программе.

В отличие от глобальных переменных, *локальные переменные* переопределяются каждый раз, когда программа выполняется, и не сохраняются в памяти. Переменная с точкой в начале имени считается локальной переменной (.pose). Локальные переменные полезны в случаях, когда несколько программ используют то же самое имя переменной.

Системы координат, схема осей и рабочая зона робота Kawasaki представлены на рисунках 3.1 и 3.5.

## 5.2 Программные инструкции (команды) движения робота

### 5.2.1 Программные инструкции.

Программные инструкции записываются в следующем формате:

Ключевое слово	параметры
JMOVE	имя переменной позиции, номер зажима

Параметр, указанный после запятой, может быть опущен. Всегда вводится пробел между ключевым словом и параметрами.

Основные программные инструкции (команды) для создания последовательности движения робота:

**JMOVE** – перемещается в интерполированном угловом движении;

**LMOVE** – перемещается в интерполированном прямолинейном движении;

**JAPPRO** – приближается к координате в интерполированном угловом движении;

**LAPPRO** – приближается к координате в интерполированном прямолинейном движении;

**JDEPART** – отходит от текущей позиции в интерполированном прямолинейном движении;

**LDEPART** – отходит от текущей позиции в интерполированном угловом движении;

**HOME** – перемещается в начальную позицию;

**C1MOVE** – перемещается с круговой интерполяцией;

**C2MOVE** – перемещается с круговой интерполяцией.

### 5.2.2 Описание отдельных команд.

Команды: **JAPPRO**, **LAPPRO**.

Форматы команд:

**JAPPRO** имя переменной позиции, расстояние;

**LAPPRO** имя переменной позиции, расстояние.

*Назначение команд.* Робот двигается по оси Z инструментальной системы координат, не доходя определенного расстояния до обученной точки. Движение

происходит: JAPPRO – по криволинейной траектории, LAPPRO – по прямолинейной траектории. Точка задается в декартовых или угловых координатах.

*Параметр имя переменной позиции* задает точку назначения в декартовых или угловых координатах.

*Параметр расстояние* – задается расстояние между обученной точкой и положением в пространстве по оси Z, не доходя до обученной точки на заданное расстояние. Если расстояние задано положительным числом, робот не доходит до точки по оси Z выше точки, если отрицательным – ниже точки.

### **Пример 1**

JAPPRO точка1, 200; криволинейное перемещение, не доходя до точки 1 на 200 мм по оси Z выше точки.

*Команды:* **JDEPART, LDEPART.**

*Форматы команд:*

**JDEPART расстояние;**

**LDEPART расстояние.**

*Назначение команд.* Манипулятор отходит от данной точки по оси Z инструментальной системы координат на определенное расстояние. Движение может быть криволинейным **JDEPART** или прямолинейным **LDEPART**.

*Параметр расстояние* задается в миллиметрах между текущей позицией и позицией, расположенной на оси Z, куда робот должен выйти. Если задано положительное значение, робот движется вверх по оси Z, если задано отрицательное, робот движется вниз по оси Z.

### **Пример 2**

LDEPART-100; перемещение по оси Z на расстояние 100 мм вниз.

## **5.3 Программные инструкции (команды) двоичных сигналов**

*Команда:* **SIGNAL.**

*Формат команды:*

**SIGNAL сигнальный номер.**

*Назначение команды.* Программируются только выходные сигналы. Включаются (ON), выключаются (OFF) заданные внешние или внутренние сигналы.

*Параметр сигнальный номер.* Задается номер внешнего выходного сигнала или внутреннего сигнала. Сигнальный номер определен, если сигнал задан как внешний выходной сигнал числами 1–32, внешние входные сигналы не задаются. Если сигнальное число положительное, сигнал устанавливается в ON, если отрицательное – в OFF.

### **Пример 3**

SIGNAL-1, 4 – выключает *первый* выход, включает *четвертый* выход на контроллере.

Внутренние сигнальные номера задаются числами 2001–2256.

**Команда: SWAIT.**

*Форматы команды:*

**SWAIT сигнальный номер.**

*Назначение команды.* Программируются в основном входные сигналы. Ожидает, пока внешний или внутренний сигнал приобретет заданное состояние.

*Параметр сигнальный номер.* Задается сигнальный номер для контроля состояния. Отрицательные числа указывают, что состояние удовлетворяется, когда сигналы выключены. Командой SWAIT сигналы задаются следующими сигнальными номерами:

- внешние выходные сигнальные номера 1–32;
- внешние входные сигнальные номера 1001–1032;
- внутренние сигнальные номера 2001–2256.

#### **Пример 4**

SWAIT-1007; внешний входной сигнал, ожидание, пока входной сигнал не выключится.

### **5.4 Программные инструкции (команды) управления скоростью перемещения звеньев робота**

Команда **SPEED** задает скорость движения робота (программную скорость).

*Формат команды:*

**SPEED speed, rotational speed, ALWAYS.**

*Параметр speed* – скорость. Выбирается программная скорость движения робота, задается от 0,01 % до 100 %, также в миллиметрах в секунду, в миллиметрах в минуту, в секундах. Если единицы измерения скорости не заданы, то скорость устанавливается в процентах.

*Параметр rotational speed* – скорость вращения.

*Параметр ALWAYS* – всегда. Если этот параметр введен, скорость остается действительной до выполнения следующей инструкции SPEED. Если параметр не введен, скорость эффективна только для следующей инструкции движения (команды).

#### **Пример 5**

SPEED 50 – устанавливает скорость следующего движения 50 %;

SPEED 100 – устанавливает скорость следующего движения 100 %;

SPEED 200 – устанавливает скорость следующего движения 100 % (скорость свыше 100 % понимается как 100 %);

SPEED 20MM/S ALWAYS – скорость движения начала инструментальной системы координат установлена 20 мм/с до тех пор, пока она не изменится другой командой SPEED, когда мониторинговая скорость 100 %.

### 5.5 Программные инструкции (команды) управления фиксаторами (схватами)

*Команда:* **OPEN, OPENI.**

*Форматы команд:*

**OPEN** номер фиксатора;

**OPENI** номер фиксатора.

*Назначение команд.* Открывает фиксаторы работа (открывает схват). Команда выдает сигналы для контролирования клапанов пневматического схвата при открытии фиксатора.

*Параметр номер фиксатора* – выбирается номер фиксатора (схвата). Если параметр пропущен, выбирается первый фиксатор.

С командой **OPEN** сигнал не выйдет до тех пор, пока не начнется следующее движение. С командой **OPENI**, если робот находится в движении, сигнал выводится после того, как движение закончено.

#### *Пример 6*

**OPEN** – сигнал открытия фиксатора послан для контроля клапана фиксатора 1, когда робот начинает следующее движение;

**OPENI 2** – сигнал открытия фиксатора послан для контроля клапана фиксатора 2, как только завершилось текущее движение.

*Команды:* **CLOSE, CLOSEI.**

*Форматы команд:*

**CLOSE** – номер фиксатора;

**CLOSEI** – номер фиксатора.

*Назначение команды.* Закрывает фиксаторы работа (закрывает схват). Команда выдает сигналы для контролирования клапанов пневматического схвата при закрытии фиксатора.

С командой **CLOSE** сигнал не выйдет до тех пор, пока не начнется следующее движение. С командой **CLOSEI**, если робот находится в движении, сигнал выводится после того, как движение закончено.

#### *Пример 7*

**CLOSE 3** – сигнал закрытия фиксатора послан для контроля клапана фиксатора 3, когда робот начинает следующее движение;

**CLOSEI** – сигнал закрытия фиксатора послан для контроля клапана фиксатора 1, как только завершилось текущее движение.

### 5.6 Мониторные команды

Команда **EDIT** запускает программу.

*Формат команды:*

**EDIT** имя программы.



**Пример 8**

>EDIT prog

Команда **E** – выход из режима редактирования и возврат в мониторный режим (конец программы Exit).

Команда: **POINT**.

Формат команды:

**POINT имя переменной позиции = значение позиции.**

*Назначение команды.* Присваивает переменной позиции (точке), имеющей имя и стоящей слева в равенстве, значение координат позиции (точки), имеющей имя и стоящей в правой части равенства.

*Параметр имя переменной позиции* определяет имя точки, которую необходимо задать в декартовых или угловых координатах.

*Параметр значение позиции* задает имя реально существующей позиции (точки). Если параметр не задан, на дисплее появляются координаты точки с запросом на изменение **Change?** При этом можно изменить координату точки, поставив в соответствующие координаты необходимые значения. Выполнение команды и ее закрытие происходит путем нажатия клавиши **Enter**.

**Пример 9**

>POINT #park

JT1	JT2	JT3	JT4	JT5	JT6
10.000	15.000	20.000	0.000	30.000	90.000

Change? (if not, hit RETURN only) (Enter)

,,,35

JT1	JT2	JT3	JT4	JT5	JT6
10.000	15.000	20.000	35.000	30.000	90.000

Change? (if not, hit RETURN only) (Enter)

>POINT pick1 = pick (Enter) – присваивает точке pick1 значение точки pick в декартовых координатах.

>POINT pos0 = #pos0 (Enter) – преобразует точку #pos0 в угловых координатах в декартовую и присваивает точке pos0 значения точки #pos0.

Команда **HERE**.

Формат команды:

**HERE имя переменной позиции.**

*Назначение команды.* Задает имя переменной позиции для текущего положения, запоминает координаты точки в пространстве. Координаты могут быть декартовыми, угловыми.

*Параметр имя переменной позиции* присваивает заданной точке имя. Имя должно начинаться с буквы или префикса #.

**Пример 10**

HERE #pick – запоминает точку в угловых координатах под именем pick.

HERE place – запоминает точку в декартовых координатах под именем place.

**5.7 Создание программы на языке AS**

Робот выполнит следующую задачу: поднимет деталь, подаваемую движущимся транспортером, и переместит ее в коробку (рисунок 5.1).

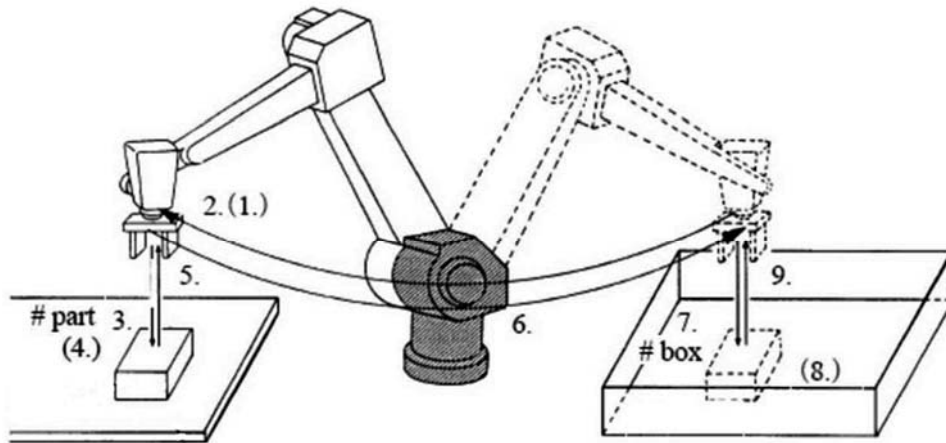


Рисунок 5.1 – Схема последовательности действий робота

**Последовательность действий, требуемых для выполнения задачи.**

- 1 Проверить, является ли схват открытым.
- 2 Переместиться в положение на 50 мм выше детали на транспортере.
- 3 Переместиться прямо вниз к детали в точку 3 (точка описывается переменной – #part).
- 4 Взять деталь и закрыть схват.
- 5 Переместиться прямо вверх на 150 мм транспортера.
- 6 Переместиться в позицию на 200 мм выше коробки.
- 7 Переместить деталь вниз в коробку в точку 7 (точка описывается переменной – #box).
- 8 Открыть схват и освободить деталь.
- 9 Переместиться снова в позицию на 180 мм выше коробки.

**Код программы**

```
>EDIT demo
.PROGRAM demo ()
1?OPENI
2?JAPPRO #part, 50
3?LMOVE #part
```

4?CLOSEI  
 5?LDEPART 150  
 6?JAPPRO #box, 200  
 7?LMOVE #box  
 8?OPENI  
 9?LDEPART 180  
 10?E

E-команда завершает режим редактирования. Когда программа выполняется, AS-система выполняет шаги в порядке от шага 1 до шага 9.

### **Задания**

#### **1 Изучить:**

- основы программирования робота;
- системы координат, схему осей и рабочую зону робота;
- программные инструкции (команды) движения робота;
- программные инструкции (команды) двоичных сигналов;
- программные инструкции (команды) управления скоростью перемещения звеньев робота;
- программные инструкции (команды) управления фиксаторами (схватками);
- мониторные команды;
- пример создания программы на языке AS;
- пример программы для обслуживания роботом технологического оборудования РТК.

2 Разработать программу для выполнения роботом транспортирования заготовок согласно варианту. Варианты заданий представлены на рисунках 5.2 и 5.3.

### ***Контрольные вопросы***

- 1 Для чего используются мониторные команды?
- 2 Для чего используются программные инструкции?
- 3 Какие возможны способы задания координат точки в трехмерном пространстве в AS-системе?
- 4 Как записываются переменные позиции и ее координаты?
- 5 В каком формате записывается каждая строка (шаг) программного языка AS?
- 6 Что такое глобальные и локальные переменные?
- 7 Какие используются системы координат при перемещении звеньев робота?
- 8 Сколько осей вращения имеет робот?
- 9 Назовите программные инструкции (команды) для программирования робота. Приведите примеры их записи.
- 10 Назовите мониторные команды. Приведите примеры их записи.
- 11 Назовите программные инструкции (команды) управления фиксаторами (схватками). Приведите примеры их записи.

12 Назовите программные инструкции (команды) управления скоростью перемещения звеньев робота. Приведите примеры их записи.

13 Назовите программные инструкции (команды) двоичных сигналов. Приведите примеры их записи.

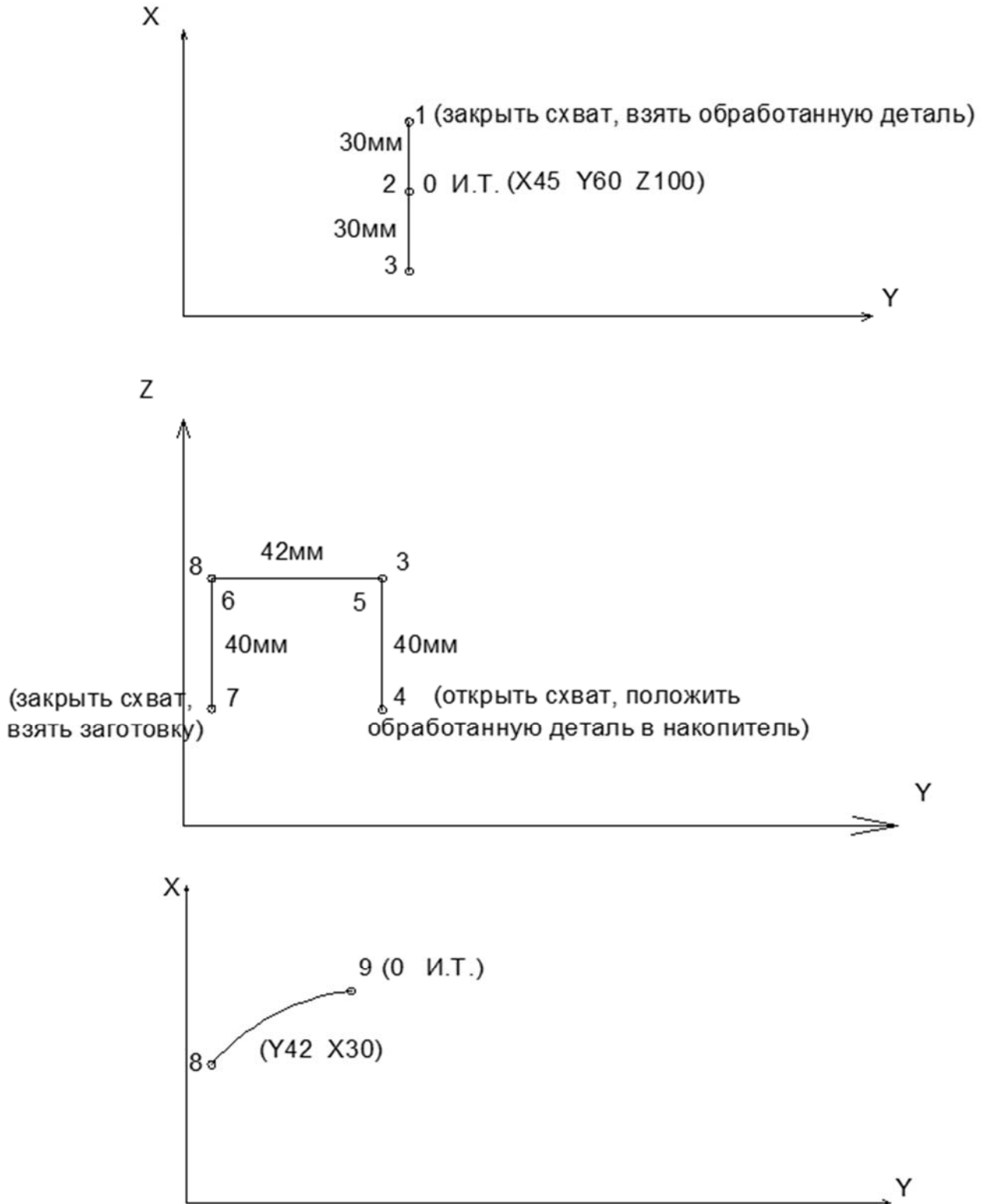


Рисунок 5.2 – Вариант 1

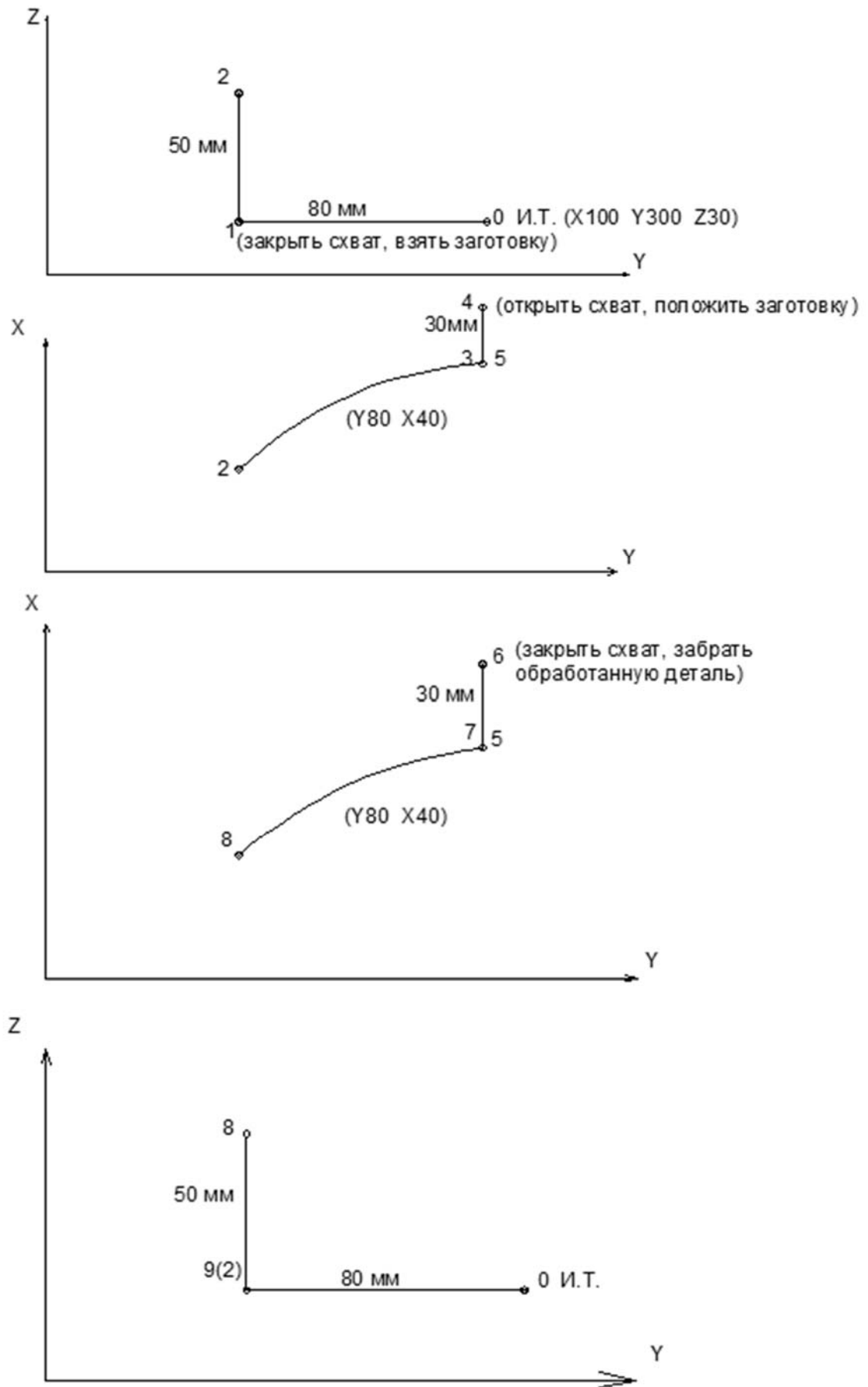


Рисунок 5.3 – Вариант 2

## 6 Практическое занятие № 5. Изучение функций ввода команд программ с помощью редактора KRterm. Отладка и выполнение программ роботом

### 6.1 Типы программ в AS-системе

Программа – это последовательность программных инструкций (команд), описывающих движение робота, функционирование внешних, внутренних сигналов, различные вычисления. Программное имя содержит не более 15 символов, начиная с алфавитной буквы, может продолжаться буквами и цифрами. Можно создать столько программ, сколько позволяет память контроллера. Программы создаются в ручном режиме (системный режим редактирования, блочное программирование) и с помощью программного обеспечения, в частности, KRterm.

### 6.2 Последовательность ввода программ

Редактор KRterm используется для создания и редактирования программ. Чтобы создать программу с именем «demo» нужно ввести «EDIT demo» и нажать клавишу Enter. Появится экран, отображающий следующее:

```
>EDIT demo
PROGRAM demo
1?
```

Теперь система ожидает ввода первого шага. Первый шаг может быть таким: вводим **OPENI** после **1?** и нажимаем Enter.

```
> EDIT demo
PROGRAM demo
1?OPENI [Enter]
2?
```

Затем можем ввести команду **JAPPRO #part, 50** для второго шага и нажимаем Enter.

```
>EDIT demo
PROGRAM demo
1?OPENI
2?JAPPRO #part, 50 [Enter]
3?
```

Вводим остальную часть программы тем же самым способом. Исправлять ошибки при вводе шагов можно, нажимая клавишу **Backspace** только перед

нажатием клавиши **Enter**. Если клавиша **Enter** нажата в конце ошибочного шага, появляется сообщение об ошибках и этот шаг бракуется. В этом случае нужно ввести шаг снова.

### 6.3 Команды редактирования

Для редактирования программ в редакторе KRterm используются команды, представленные в таблице 6.1.

Таблица 6.1 – Команды редактора

Название команды	Описание команды
EDIT	Запускает программу редактора
C	Заканчивает редактирование текущей программы и заменяет ее другой программой (Change)
S	Выбирает программный шаг для отображения (Step)
P	Отображает заданное количество программных шагов (Print)
L	Выбирает предыдущий шаг (Last)
I	Вставляет новый шаг (Insert)
D	Удаляет программные шаги (Delete)
R	Заменяет символы (Replace)
O	Размещение курсора в текущем шаге (One line)

### 6.4 Описание команд редактирования

Команда **C**.

*Формат команды:*

**C имя программы, номер шага**

*Назначение команды.* Заменяет программу, находящуюся в редакторе на другую.

*Параметр имя программы* определяет программу для редактирования.

*Параметр номер шага* выбирает номер шага, с которого начинается редактирование. Если номер шага не задан, редактирование начинается с первого шага.

Команда **S**.

*Формат команды:*

**S номер шага**

*Назначение команды.* Выбирает и отображает шаг программы, выбранный для редактирования.

*Параметр номер шага.* Если номер шага не задан, происходит переход на первый шаг программы. Если номер шага больше чем шагов программы, происходит переход на шаг, следующий за последним шагом программы.

**Команда P.***Формат команды:***P количество шагов**

*Назначение команды.* Отображает число шагов программы, заданных в параметре, начиная с текущего шага. Последний шаг в списке готов к редактированию.

*Параметр количество шагов* задает количество шагов для просмотра. Если количество шагов не задано, отображаются все шаги, следующие за текущим шагом.

**Команда L.**

*Назначение команды.* Отображает предыдущий шаг.

**Команда I.**

*Назначение команды.* Вставляет шаг перед текущим шагом. Заканчивается ввод команды нажатием клавиши **Enter**. Шаги после вставленной строки перенумеровываются. Все строки, вставленные перед выходом из режима вставки (нажатие клавиши **Enter**), вставлены в программу.

**Пример 1**

CLOSEI команда вставлена между шагами 3 и 4.

```

1?OPENI
2?JAPPRO #PART, 500
3?LMOVE #PART
4?LDEPART 1000
5?S 4; отображает шаг 4 для вставки шага перед ним
4LDEPART 1000
4?I ; Type the I command (введите команду I)
4I CLOSEI; вставляем команду CLOSEI
5I [Enter]; нажимаем ввод для завершения вставки шагов
5 LDEPART 1000; step 4 (шаг 4-ый стал пятым)
5?

```

**Команда D.***Формат команды:***D количество шагов**

*Назначение команды.* Удаляет заданное число шагов, следующих за текущим шагом.

*Параметр количество шагов.* Задает число шагов для удаления, начиная с текущего шага. Если заданное количество шагов больше шагов программы, удаляются все шаги, следующие за текущим шагом. После удаления все оставшиеся шаги автоматически перенумеровываются и сдвигаются вверх.



Команда **R**.

*Формат команды:*

**R** *символьная цепочка*

*Назначение команды.* Заменяет ранее введенные символы в данном шаге на последовательность заданных символов.

*Параметр* **символьная цепочка** задает новые символы для замены ранее введенных символов.

Последовательность действий для использования команды **R** следующая.

1 Используя клавишу пробел **Spacebar**, установить курсор под первым символом, который необходимо изменить.

2 Нажать клавишу **R** и затем **Spacebar**.

3 Ввести новый символ замены.

4 Нажать клавишу **Enter**.

### **Пример 2**

Изменить скорость движения с 20 % на 35 %, используя команду **R**.

1 SPEED 20, ALWAYS

1? R 35 [Enter]

1 SPEED 35, ALWAYS

1?

Команда **O**.

*Назначение команды.* Устанавливает курсор за последним символом выбранного шага.

### **Пример 3**

Изменить имя точки aa1 на cc1, используя команду **O**.

3 JMOVE aa1

3?O

3 JMOVE aa1

3 JMOVE aa1 [BackSpace]; удаляем aa1

3 JMOVE cc1 [Enter]; вводим cc1

3?

## **6.5 Запуск редактора KRterm (терминала) для ввода команд**

Запись программы в виде команд (программных инструкций и мониторинговых команд) выполняется в следующей последовательности.

1 Открыть программу (терминал) KRterm.

2 Подключиться к контроллеру. В окне (рисунок 6.1) выбрать Com/имя (первая строчка).

3 Ввести логин.

4 Ввести команду EDIT имя программы.

- 5 Ввести команды – программные инструкции.
- 6 Ввести мониторинговые команды.
- 7 Выполнить введенную программу.

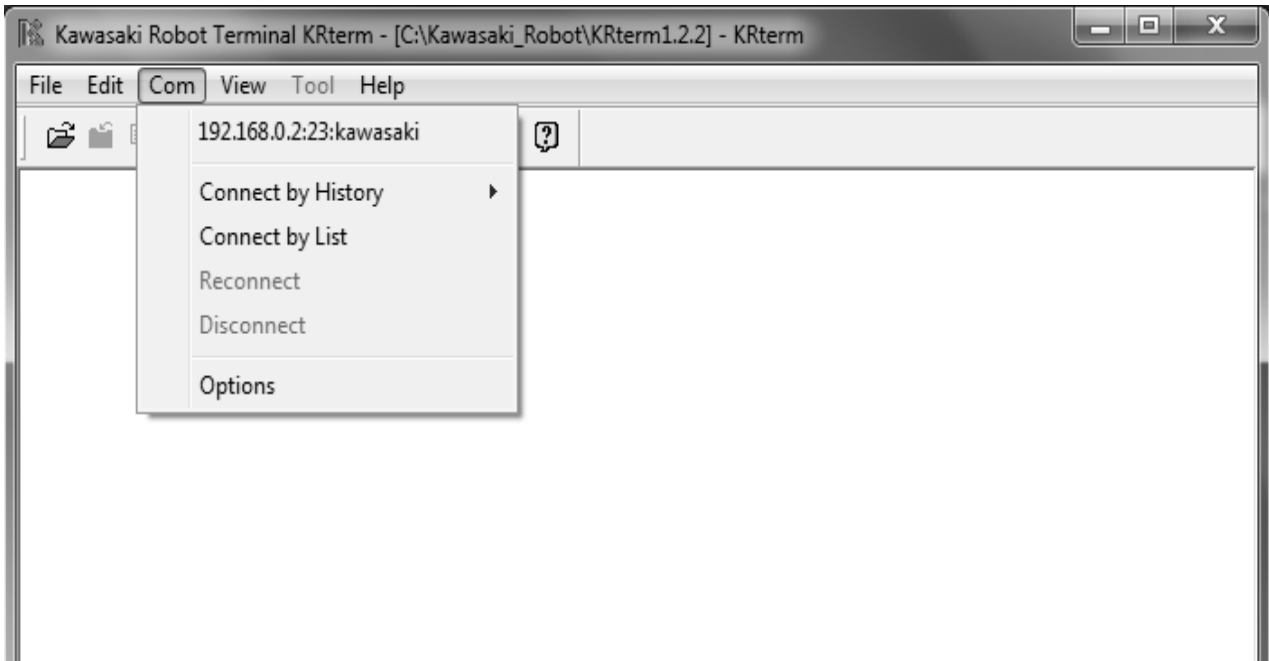


Рисунок 6.1 – Окно терминала редактора KRterm

### ***6.6 Выполнение управляющих программ робота***

После ввода команд программы в редакторе KRterm программа автоматически записывается в память контроллера. Редактор с кодом введенной программы можно не закрывать при ее выполнении.

Для выполнения программы нужно установить переключатель **TEACH/REPEAT** на контроллере в положение **REPEAT**. Переключатель **TEACH LOCK** на пульте ручного управления должен находиться в положении **OFF**. Затем нужно включить силовое питание привода на пульте ручного управления нажатием клавиш **A + Motor ON** и **A + RUN** (рисунок 2.5).

Мониторная скорость должна быть ниже 30 %. Вначале нужно установить скорость 10 %.

```
>SPEED 10 [Enter]
```

Скорость можно установить и в ручном режиме нажимая клавиши **A + SPEED**. Если робот движется корректно со скоростью 10 %, то ее можно увеличить. Если движения робота выполняются некорректно, то нужно нажать клавиши пульта ручного управления **A + HOLD**. Робот замедлится и остановится. В случае аварийной ситуации нужно нажать кнопку аварийного останова **EMERGENCY STOP** (красного цвета) на панели контроллера или на пульте ручного управления. Тормозная система робота сработает, и робот остановится

немедленно.

Для того чтобы начать выполнение программы, можно использовать команду EXECUTE.

>EXECUTE demo [Enter]

Робот должен выполнить программу.

>SPEED 30 [Enter]

>EXECUTE demo [Enter]; *робот движется со скоростью 30 %.*

После того как программа EXECUTE была выполнена не менее одного раза, можно использовать клавиши **A + Cycle Start** пульта ручного управления.

Чтобы выполнить программу не один раз, нужно ввести номер повторений после имени программы.

EXECUTE demo,5 [Enter]; *выполняет программу 5 раз*

EXECUTE demo,-1 [Enter]; *выполняет программу непрерывно*

Выполнение программы через мониторную команду STEP или клавишу пульта управления **ЧЕКС** дает возможность проверить движение и содержимое программы при выполнении шаг за шагом.

При записи команд в редакторе KRterm координаты точек можно вводить, используя мониторные команды POINT и HERE, или обучить точки в ручном режиме. При обучении точек нужно подвести схват робота к требуемой точке в пространстве в ручном режиме, потом ввести в редакторе KRterm:

>HERE имя точки [Enter]

Координаты точек появятся на экране.

Выполнение программы можно осуществлять и в ручном режиме. Для этого нужно выбрать программу, используя пульт ручного управления. Нужно выбрать из пошагового меню функцию Directory (рисунок 6.2). После этого появится экран выбора программы (рисунок 6.3) и из списка следует выбрать программу.

Затем нужно запустить программу, установив переключатель **TEACH/REPEAT** на контроллере в положение **REPEAT**, переключатель **TEACH LOCK** на пульте ручного управления – в положение **OFF** и нажать клавиши **A + Motor ON**, **A + RUN** и **A + Cycle Start**.

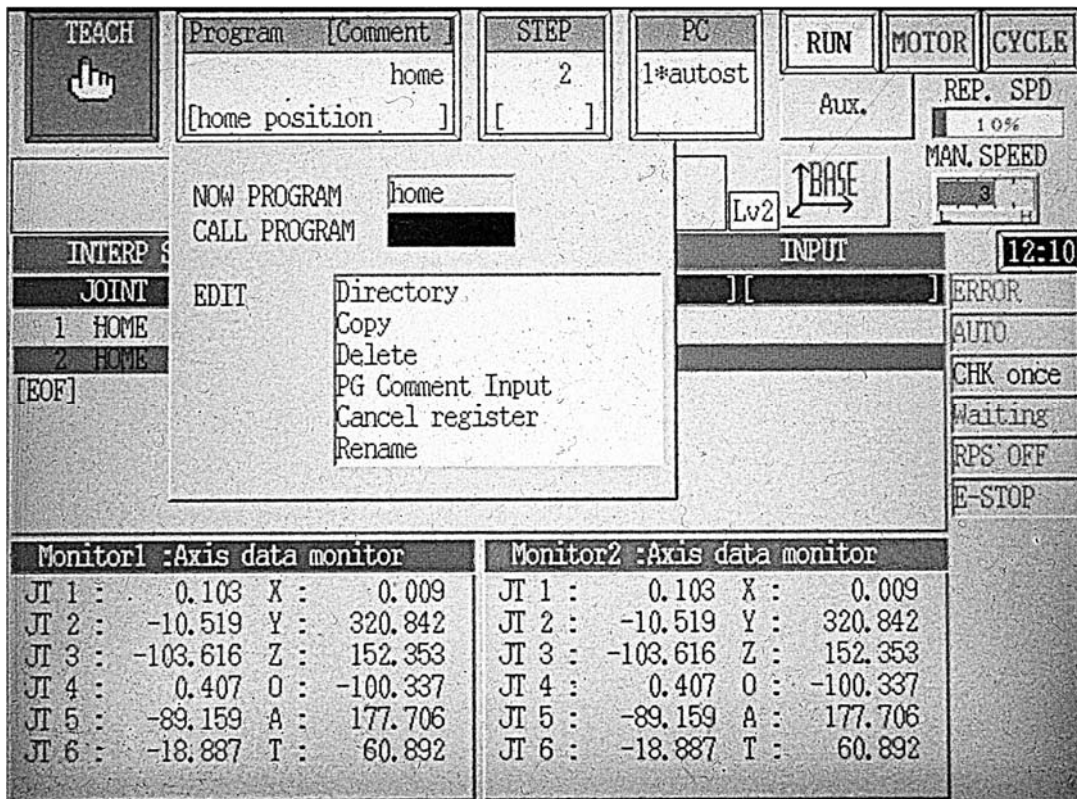


Рисунок 6.2 – Экран выбора функции Directory на пульте ручного управления

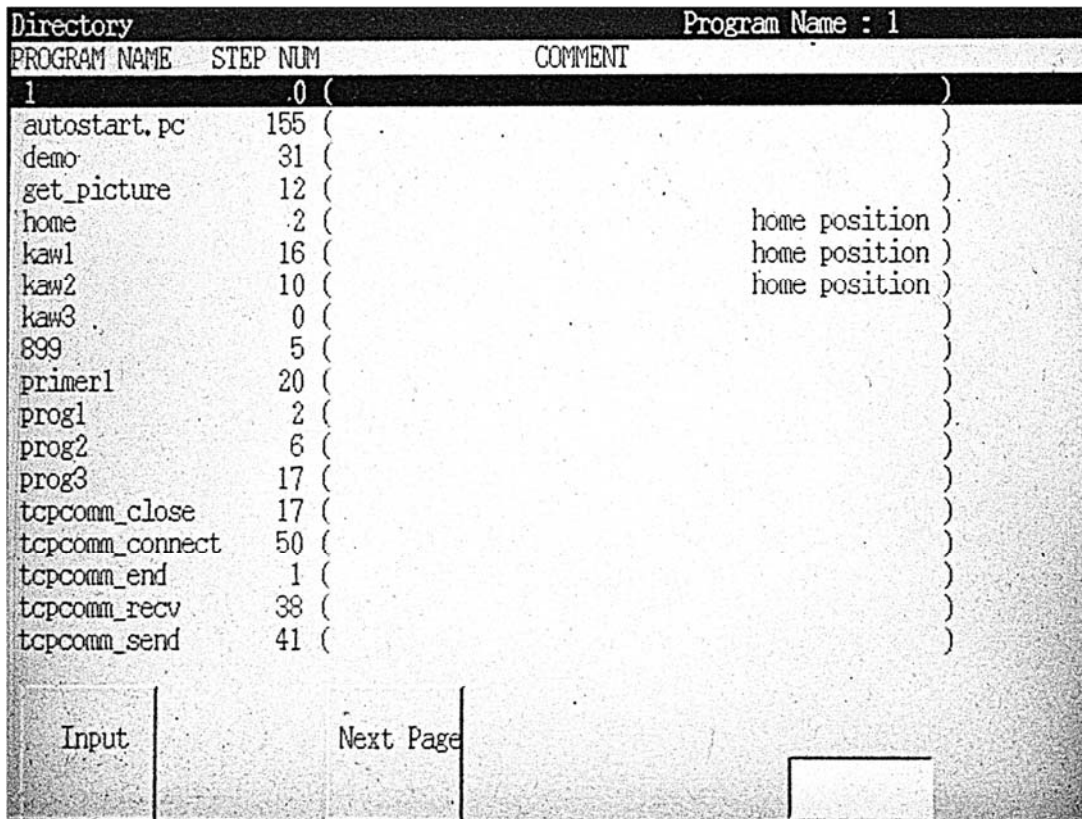


Рисунок 6.3 – Экран Directory выбора программы на пульте ручного управления

## ***6.7 Останов выполнения программ***

Существует несколько способов остановки выполнения программы во время работы робота. Ниже приведены три способа останова от самого срочного до менее срочного.

1 Нажимаем кнопку аварийного останова (красного цвета) **EMERGENCY STOP** на операционной панели контроллера или на пульте ручного управления. Тормозная система сработает и робот немедленно остановится. До тех пор, пока нет аварийной ситуации, нужно использовать способы 2 или 3.

2 Нажимаем клавиши пульта ручного управления **A + HOLD**. Робот замедлится и остановится.

3 Ввод команды **ABORT** останавливает программное выполнение после того, как робот завершит текущий шаг (инструкцию движения).

>ABORT [Enter]

Для останова выполнения программы также может быть использована команда **HOLD**.

>HOLD [Enter]

## ***6.8 Возобновление выполнения управляющих программ робота***

В зависимости от того, каким способом робот был остановлен, существует несколько способов возобновления выполнения программ.

1 Если робот был остановлен при помощи кнопки аварийного останова (красного цвета) **EMERGENCY STOP**, то нужно разблокировать эту кнопку и нажать клавиши пульта ручного управления **MOTOR POWER** для того, чтобы включить силовое питание привода. Робот начинает движение после нажатия клавиш **A + Cycle Start**.

2 Если для останова робота были использованы клавиши пульта ручного управления **A + HOLD**, то нужно нажать клавиши **A + RUN** для продолжения выполнения программы.

3 Для того чтобы продолжить выполнение программы после команд **ABORT** или **HOLD**, или когда выполнение программы было прервано из-за какой-то ошибки, нужно использовать команду **CONTINUE** (ошибка должна быть сброшена до возобновления выполнения программы).

>CONTINUE [Enter]

***Пример*** программы, записанной в редакторе KRterm

```

This is AS monitor terminal "AUX1"
>EDIT prog3
  .Program prog3()
  1?HOME
  2?SPEED 50
  3?LMOVE tochka4
  4?LMOVE tochka5
  5?LMOVE tochka6
  6?SPEED 100
  7?LDEPART -200
  8?SPEED 30
  9?JMOVE tochka7
  10?JAPPRO tochka6, -200
  11?SPEED 100
  12?LAPPRO tochka6, 70
  13?SPEED 20
  14?LMOVE tochka6
  15?DELAY 5
  16?SPEED 100
  17?E
>POINT tochka4
  X[mm]      Y[mm]      Z[mm]      O[deg]      A[deg]      T[deg]
  0.000      0.000      0.000      0.000      0.000      0.000
Change? (If not, Press RETURN only.)
Program held.No = 1
-220,450,130,-100,177,61
  X[mm]      Y[mm]      Z[mm]      O[deg]      A[deg]      T[deg]
  -220.000   450.000   130.000  -100.000   177.000    61.000
Change? (If not, Press RETURN only.)

>POINT tochka5
  X[mm]      Y[mm]      Z[mm]      O[deg]      A[deg]      T[deg]
  0.000      0.000      0.000      0.000      0.000      0.000
Change? (If not, Press RETURN only.)
200,300,200(D1561) [Power sequence board]AC primary power OFF.

PC program aborted.No = 1001
Connecting to Kawasaki E Controller

```

## Задания

### 1 Изучить:

- последовательность ввода программ;
- команды редактирования;
- последовательность выполнения управляющих программ работа;
- последовательность останова выполнения программ;
- последовательность возобновления выполнения управляющих программ работа;
- команды управления программой и данными;
- команды сохранения программ и данных.

2 Выполнить запись программы в виде команд (программных инструкций и мониторных команд).

3 Отредактировать программу и выполнить введенную программу.

### **Контрольные вопросы**

- 1 Назовите типы программ в AS-системе.
- 2 Какова последовательность ввода программ в терминале?
- 3 Назовите команды редактирования. Приведите примеры.
- 4 Какова последовательность запуска редактора KRterm (терминала) для ввода команд?
- 5 Какова последовательность выполнения управляющих программ робота?
- 6 Назовите способы останова выполнения программы.
- 7 Как осуществляется возобновление выполнения управляющих программ робота?
- 8 Назовите команды управления программой и данными.
- 9 Назовите команды сохранения программ и данных.

### **Список литературы**

- 1 **Егоров, О. Д.** Конструирование механизмов роботов: учебник / О. Д. Егоров. – Москва: Абрис; Высшая школа, 2012. – 444 с.: ил.
- 2 **Клепиков, В. В.** Технологическая оснастка. Станочные приспособления: учебное пособие / В. В. Клепиков. – Москва: ИНФРА-М, 2019. – 345 с.
- 3 **Иванов, А. А.** Основы робототехники: учебное пособие / А. А. Иванов. – Москва: ФОРУМ, 2015. – 224 с.
- 4 Технологическая оснастка: учебник / М. Ф. Пашкевич [и др.]; под ред. М. Ф. Пашкевича. – Минск: Адукацыя і выхаванне, 2002. – 320 с.
- 5 Технологическая оснастка: учебное пособие для вузов / В. Е. Антонюк [и др.]; под ред. В. Е. Антонюк. – Минск: Изд-во Гревцова, 2011. – 376 с.
- 6 Справочник технолога-машиностроителя: в 2 т. / Под ред. А. Г. Косиловой и Р. К. Мещерякова. – Москва: Машиностроение, 1985. – Т. 1. – 656 с.