



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
УПРАВЛЕНИЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ И ПОВЫШЕНИЯ
КВАЛИФИКАЦИИ

Кафедра «Робототехника и мехатроника»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению лабораторных работ

по дисциплине

«Применение мехатронных систем»

Автор

Мироненко Р.С.

Ростов-на-Дону, 2013





Аннотация

Лабораторные занятия по дисциплине “Применение мехатронных систем” посвящены программированию промышленных роботов для выполнения ими технологических функций (например, сборки узла механической подсистемы мехатронных систем) с использованием компьютерных эмуляторов.

В процессе проведения лабораторных занятий используются компьютерный эмулятор промышленного робота PM-01 на основе системы программирования ARPS, компьютерный эмулятор промышленного робота KUKA IR 161/15 на основе системы программирования SRCL, компьютерный эмулятор МПЦУ.

Цель лабораторных работ заключается в ознакомлении с системами программирования ARPS и языком программирования SRCL, а также в получении навыков программирования МПЦУ и промышленных роботов PM-01 и KUKA IR 161/15.

Автор

к.т.н., доц. Мироненко Р.С.





Оглавление

Описание системы программирования ARPS.....	4
Команды, реализованные в компьютерном эмуляторе.....	13
Порядок выполнения работы.....	15
<i>Отчет по лабораторной работе должен содержать.....</i>	15
Контрольные вопросы ориентированы на проверку знаний студента в области:	16
Устройство МПЦУ	17
Режимы работы МПЦУ	18
Система команд МПЦУ	20
Формат команды МПЦУ.....	21
Программирование МПЦУ в режиме эмуляции	25
Задание на вариант №1	26
Задание на вариант №2	27
Задание на вариант №3	28
Задание на вариант №4	29
Описание системы программирования SRCL	31
Порядок выполнения работы.....	31
Контрольные вопросы ориентированы на проверку знаний студента в области:	32



ОПИСАНИЕ СИСТЕМЫ ПРОГРАММИРОВАНИЯ ARPS

Система программирования ARPS является основой программного обеспечения промышленного робота РМ-01 и позволяет осуществлять программирование работы робота методом обучения и аналитически, путем задания последовательных положений звеньев, либо координат схвата. ARPS позволяет запись готовых программ на отдельный носитель информации, составление новых программ в процессе работы робота, формирование библиотеки подпрограмм. Пользователю доступна также особая группа команд для выполнения специальных операций и редактирования текста, а также команды пультового терминала, позволяющие взаимодействовать с системой на уровне машинных кодов.

Язык программирования ARPS допускает использование следующих типов переменных и констант: целые, действительные, переменные положения. Для обозначения переменных используются буквы А...Z, цифры 0...9 и знак точки ".". Длина имени не ограничена, но не рекомендуется использовать более 10 знаков. Первым символом имени всегда должна быть буква.

Целые числа используются для обозначения каналов ввода-вывода, количества повторений цикла, номеров звеньев манипулятора и т.п. Они могут изменяться в пределах от -32768 до +32767. Над целыми числами можно производить арифметические действия.

Действительные числа используются при определении расстояний, угловых величин, скорости, времени задержки. Расстояния задаются в миллиметрах и могут принимать значения от -1024,00 до +1023,99 мм с дискретностью 0,01 мм. Угловые величины задаются в градусах и изменяются от -180 до +179,995° с дискретностью 0,005°. Скорость перемещения манипулятора устанавливается заданием абсолютных значений (в мм/с) и коэффициентом масштабирования (в %) и изменяется в пределах от 2,0 до 3000,0 мм/с с дискретностью 0,1 мм/с. Время задержки задается в секундах и может изменяться от 0,00 до 327,7 с дискретностью 0,01 с.

Все команды языка ARPS по функциональному назначению можно разделить на следующие группы:

- команды определения точек;
- команды установки скорости;
- команды управления движением манипулятора;



- команды управления схватом;
- команды передачи управления в программе;
- команды управления входами/выходами;
- команды сдвига координат инструмента;
- команды управления конфигурацией руки робота;
- команды останова;
- команды управления НГМД;
- специальные команды.

Команды определения точек

Точка определяет ту позицию в технологической зоне робота, в которую (или через которую) манипулятор должен переместиться, а также ориентацию схвата. Различают три типа точек: абсолютные, координатные и комбинированные. Точке присваивается имя. В качестве имени используется совокупность символов из букв /A-Z/, цифр /0-9 / и знака "#", не превышая 10 знаков. Первым символом имени точки должна быть буква или символ "#" - для абсолютных точек.

Абсолютные точки определяют положение манипулятора в обобщенной системе координат. Координатные точки определяют положение манипулятора в базовой (декартовой) системе координат. Комбинированные точки формируются относительно какой-нибудь ранее определенной точки, которая должна быть координатной.

Команда HERE служит для фиксирования в 3У координат текущей позиции манипулятора для указанной в команде точки.

Формат команды: HERE <точка>, где <точка> - имя абсолютной, координатной или комбинированной точки. Задание координат производится выводом манипулятора в требуемое положение с помощью пульта ручного управления и ввода после этого команды HERE.

Команда FRAME служит для формирования рабочей плоскости для робота. Она имеет формат: FRAME S=A,AX,PX[AO], где S - название формируемой плоскости; A - точка начала системы координат в FRAME; AX - направление оси X FRAME относительно точки A; PX - точка, через которую проходит плоскость, параллельная оси X FRAME; AO - окончательное начало системы координат FRAME: значения X, Y, Z. Если AO не задано, то начало системы координат расположено в точке A.

Команда DISTANCE служит для вычисления расстояния между двумя точками. Формат команды: DISTANCE S=T1,T2, где



S-целочисленная переменная, которой присваивается значение расстояния между точками T1 и T2, которые должны быть координатными.

Команда LOCATE служит для определения точки через другую. Формат команды: LOCATE T1=[INVERSE]T2. Например: LOCATE #A1=#A2 - абсолютная точка #A1 принимает значение координаты точки #A2.

Команда SHIFT служит для перемещения точки в основной системе координат. Она имеет формат: SHIFT T=[dx],[dy],[dz], где dx - приращение в мм, прибавляемое к значению x точки T; dy - приращение, прибавляемое к y; dz - приращение, прибавляемое к z.

Команды установки скорости

С помощью этих команд задаются скорости движения инструмента, закрепленного на манипуляторе.

Командой SPEED задается базовая скорость движения инструмента. Формат команды: SPEED V, где V-скорость в мм/с по прямой (V= 2...3000 мм/с). Например, SPEED 4000 задает скорость движения инструмента, равную 400 мм/с.

Команда SPEED NEXT задает базовую скорость движения инструмента только на время следующей команды движения.

Командой SPEED % задается масштабный коэффициент скорости. При выполнении команд перемещения манипулятора действительная скорость движения инструмента получается путем умножения базовой скорости на масштабный коэффициент, задаваемой целой переменной или целой константой в диапазоне 1...300%. Например, SPEED %50 устанавливает масштабный коэффициент скорости, равный 50%. Если базовая скорость была 200 мм/с, то после выполнения команды действительная скорость будет равна 100 мм/с.

Командой MAX SPEED задается верхний (максимальный) предел скорости движения манипулятора. По умолчанию верхний предел скорости принимается равным 500 мм/с.

Команды управления схватом

К этим командам относятся: OPEN, CLOSE, ODELAY, CDELAY.

Команда OPEN вызывает разжатие схвата (время разжатия 0,2 сек).

Командой CLOSE реализуется сжатие схвата. По умолчанию время задержки при сжатии составляет 0,2 сек.



С помощью команды ODELAY устанавливается время задержки схвата при разжатии. Формат команды: ODELAY T, где T - время задержки на разжатие схвата.

Командой CDELAY устанавливается время на сжатие схвата.

Команды управления входами/выходами

С помощью этих команд осуществляется перевод соответствующих линий входов/выходов в активное или пассивное состояние, проверка состояния линий, управление разрешением/запрещением прерываний.

Команда OUT управляет состоянием выходных линий.

С помощью команды RUNOUT в пассивное состояние переводятся все отмеченные линии.

Командой OUTGROUP устанавливается значение указанного 16-канального выходного порта.

С помощью команды WAITIN можно проверить состояние соответствующей линии. При выполнении этой команды происходит ожидание до тех пор, пока на всех указанных линиях не будет достигнуто требуемое состояние.

Командой INCALL инициируется внешнее прерывание.

Команды сдвига системы координат инструмента

Команда TOOL имеет формат: TOOL [dx],[dy],[dz],[δx],[δy],[δz], где dx, dy, dz - величины сдвига в мм по осям x,y,z соответственно, δx , δy , δz - величины сдвига в градусах по соответствующим осям.

Команда LTOOL имеет формат: LTOOL T, где T - координатная точка, значение которой задается в качестве сдвига системы координат инструмента.

Команды управления конфигурацией руки робота

Командой J2 RIGHT выбирается конфигурация "правая рука" (при движении манипулятора в следующую точку, шарнир 2 находится на правой стороне при взгляде сзади кожуха данного шарнира).

Командой J2 LEFT выбирается конфигурация "левая рука".

Командой J3 UP выбирается конфигурация



"локоть в правом положении", а командой J3 DOWN - "локоть в левом положении" (для шарнира 3).

С помощью команд J5 PLUS и J5 MINUS задается конфигурация пятого шарнира: угловая величина положительная или отрицательная.

Команды останова

Команда STOP имеет формат: STOP ["текст"],[S], где S - число; "текст" - произвольный текст. При выполнении команды STOP прекращается выполнение программы и на экран отображается текст и число. Продолжение выполнения программы возможно после ввода директивы CONTINUE.

Команда HALT, имеющая формат: HALT ["текст"],[S] идентична команде STOP, за исключением того, что после этой команды невозможно продолжить выполнение программы директивой CONTINUE.

По команде EXIT останавливается выполнение программы при окончании рабочего цикла.

Команды управления НГМД

Команда LOAD используется для загрузки в ЗУ из НГМД программ или точек.

Команда DLOAD используется для удаления программ и/или точек из файлов, размещенных предварительно в ЗУ.

Команда FRACK предназначена для подготовки к использованию свободных зон НГМД, образовавшихся за счет сжатия файлов.

Команда STORE предназначена для записи программ и/или точек, находящихся в данный момент в ЗУ на НГМД с соответствующим именем.

Операцию записи на НГМД можно прервать вводом команды CONTROL/S.



Команды передачи управления в программе

Команда JUMP используется для безусловной передачи управления.

Команда IF применяется для организации ветвлений.

Команда CALL используется для передачи управления подпрограмме.

Каждая подпрограмма должна завершаться командой RETURN. Эта команда используется для возврата управления из подпрограммы в вызывающую программу.

Специальные команды

Командой BASE производится сдвиг основной системы координат.

Команда DELAY используется для программного задания временных задержек.

Команда DISABLE используется для вычисления программного прерывателя.

Команда SET используется для вычисления значения переменной целого типа.

По команде CAL выполняется калибровка манипулятора, т.е. вычисляются точные углы сочленений шарниров.

Командой CLOAD производится загрузка новых калибровочных данных с диска для энкодера двигателей манипулятора.

Командой ZERO указанные линии выхода устанавливаются в пассивное (нулевое) состояние.

Команда C /текст/ выводит соответствующую строку текста.

Команда EDIT PROG инициирует редактирование программы.

Команды управления движением манипулятора

Командами управления движением манипулятора осуществляется его перемещение в заданную точку.

Командой GO реализуется интерполированная траектория движения в заданную точку. Формат команды GO T, где T - абсолютная, координатная или комбинированная точка.

Командой GOS реализуется прямолинейное движение из текущей точки в заданную.



Робототехника и мехатроника

Командами GO OPEN, GOS OPEN выполняются операции, аналогичные вышеописанным командам, но в начале движения выполняется разжатие схвата манипулятора.

Команды GO CLOSE и GOS CLOSE аналогичны командам GO и GOS, но в начале движения выполняется сжатие схвата.

Командами GO NEAR и GOS NEAR реализуется перемещение манипулятора по интерполированной /прямолинейной/ траектории на требуемое расстояние к заданной точке в направлении оси системы координат инструмента.

Команда GOS WEAVE предназначена для перемещения манипулятора по прямолинейной траектории в заданную точку при одновременном колебании манипулятора.

По командам MOVE или MOVES манипулятор перемещается на заданное расстояние относительно основной системы координат соответственно по интерполированной или прямолинейной траектории.

По командам TMOVE или TMOVES манипулятор перемещается на заданное расстояние относительно системы координат инструмента по интерполированной или прямолинейной траектории.

С помощью команды MOVE JOINT выполняются движения отдельных шарниров робота. Формат команды: MOVE JOINT N, F, где N - целочисленная константа от 1 до 6, указывающая номер шарнира; F - величина угла в градусах, на который необходимо переместить шарнир.

По команде ALIGN происходит выравнивание инструмента манипулятора по осям основной системы координат.

По команде GO READY манипулятор перемещается в вертикальное положение.

Переключатели программ

С помощью переключателей (фиксированные переменные) можно программно управлять режимами работы системы ARPS. Переключатели могут быть в одном из двух возможных состояний: активном или пассивном. Смена состояний переключателя возможна с помощью команд: ENABLE (перевести в активное состояние) и DISABLE (перевести в пассивное состояние).

В системе ARPS возможно использование следующих переключателей: BREAK, DISTIO, INCALL, PRINT, SEPPOR.

Переключатель BREAK влияет на непрерывность работы



робота. Переключатель DISTIO влияет на режим выполнения команды DISTANCE. Переключатель INCALL влияет на разрешение/запрещение прерывания по входным линиям 1...16. Переключатель PRINT влияет на режим выполнения команды PRINT. Переключатель SEPPOP влияет на контроль ошибки сервопривода.

Команды определения точек

WERE выводит на экран значения координат текущей позиции инструмента манипулятора.

Командой LTEACH реализуется обучение робота в режиме ручного управления.

Командой CHANGE реализуется вывод значения точки на экран дисплея с одновременным запросом на изменение.

Команды выполнения программы

По команде RUN PROG или RUN PROG,N запускается на выполнение загруженная в ЗУ программа PROG (N - число повторений программы).

Команда EXIT позволяет остановить выполнение программы в конце рабочего цикла. При этом число повторений обнуляется.

Команда ABORT прерывает выполнение программы в конце текущего шага.

Командой CONTINUE возобновляется выполнение прерванной программы.

Команды удаления

По команде LDEL из ЗУ стираются координаты указанных точек, которые могут быть абсолютными или координатными.

Команда FDEL служит для удаления файла с НГМД.

Командой DLOAD производится считывание из НГМД в ЗУ файла и удаление из ЗУ блока управления текущих программ и/или точек, которые имеются в этом файле.

По команде ZERO DISK удаляется содержимое НГМД.



Команды получения листингов

По команде PLIST PROG на дисплей выводится текст программы PROG, а по команде PLIST выводятся тексты всех программ, хранящихся в ЗУ. Командой LLIST выводятся все точки, хранящиеся в ЗУ, а по команде LLIST T1,T2,... на экран выводятся точки T1,T2,....

Командой FDIR на экран выводится список имен файлов, их длина в блоках (объем блока 256 байт) и объем свободных участков НГМД в блоках.

Команда LIST COMANS выводит на дисплей в сокращенной форме команды монитора и редактора, а также типы аргументов команд.

Команда LIST STATUS выводит на дисплей данные о состоянии системы: сдвиг основной системы координат; объем свободной области памяти в ОЗУ; состояние переключателя программ; сдвиг системы координат инструмента; выполняемая программа и шаги программы; выполненное число программ и общее число прогонов; задание скорости (максимальная скорость, абсолютная скорость, масштабный коэффициент); зеркальная система координат (если используется).

Команда SKALE выполняет масштабирование координат x, y, z .



КОМАНДЫ, РЕАЛИЗОВАННЫЕ В КОМПЬЮТЕРНОМ ЭМУЛЯТОРЕ

Команды определения точек

WHERE

Команда позволяет вывести на экран текущие координаты манипулятора.

Если указан символ #, выводятся абсолютные координаты, в противном случае декартовы координаты схвата. Остановка работы команды производится клавишами CTRL-"S"

Команды задания точек

HERE #<Имя точки>

Команда позволяет задать абсолютную координатную точку с указанным именем.

Команды управления движением манипулятора

GO #<Имя точки>

Команда движения схвата в определенную точку.

SPEED <скорость>

Команда задания скорости движения. Допустимый диапазон (2...3000).

Команды управления схватом

OPEN

Команда открытия схвата.

CLOSE

Команда закрытия схвата.

ODELAY <время>

Команда открытия схвата с временной задержкой. Время задается в 0,1 секунды.

CDELAY <время>

Команда закрытия схвата с временной задержкой.



Команды редактирования программы

EDIT <имя программы>

Начинает редактирование программы. Если в памяти находится ранее набранная программа, то она удаляется. Остановка работы команды производится клавишами CTRL-"S".

PLIST

Выводит на экран программу, находящуюся в памяти.

Специальные команды

RUN

Запускает программу, находящуюся в памяти.

PRINT <текст>

Распечатывает указанный после команды текст.

DELAY <время>

Осуществление задержки.

PSAVE <имя>

Запись набранной в режиме редактирования программы в файл с указанным именем.

PLOAD <имя>

Чтение ранее сохраненной программы из файла с указанным именем.



ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. В отчете нарисовать схему компоновки РТК, в состав которого, помимо ПР РМ01, входит такое технологическое оборудование, как, например, металлообрабатывающие станки, бункерно-загрузочные устройства для заготовок или готовых изделий, конвейерные линии и т.д.

2. Написать алгоритм работы робота для созданного РТК.

3. Реализовать полученный алгоритм работы робота посредством программирования с помощью компьютерного эмулятора.

Для начала работы с эмулятором необходимо запустить файл RM 2000.exe и через раздел программы Views открыть дополнительные окна, в которых изображен пульт оператора ПР РМ-01 (Manual control) и визуализируется модель робота (Render window).

Программирование робота осуществляется в следующей последовательности:

1) обучение позициям (точкам) (общее количество должно быть не менее 12), в которую (или через которую) должен переместиться схват робота в соответствии с разработанной схемой РТК. Координаты каждой точки фиксируются в памяти эмулятора посредством команды HERE. Эмулятор позволяет определять только абсолютные точки. Перемещение в требуемую точку осуществляется с помощью пульта оператора, в котором расположение и названия "кнопок" и "индикаторов" соответствуют оригинальному пульта ручного управления ПР РМ-01;

2) написание программы. Ввод программы начинается с команды edit <имя программы>. В тексте программы используются имена точек, созданных на первом этапе. При написании программы необходимо обеспечить появление в процессе ее отработки комментариев по поводу каждого действия робота. Также требованием к программе является использование всех команд, реализованных в эмуляторе;

3) отладка программы.

Отчет по лабораторной работе должен содержать:

1. Фамилию студента, название группы, дату сдачи.
2. Название работы, ее цель.
3. Схему компоновки РТК.



4. Алгоритм работы робота.
5. Текст программы.
6. Координаты точек, используемых программой.

Контрольные вопросы ориентированы на проверку знаний студента в области:

1. общих сведений о системе программирования ARPS;
2. используемых в данном языке команд: их назначение, формат (для команд, реализованных в эмуляторе);
3. порядка программирования ПР РМ01.



УСТРОЙСТВО МПЦУ

МПЦУ предназначено для управления РТК и автоматическими линиями при автоматизации технологических циклов в условиях средне и мелкосерийного производства.

МПЦУ представляет собой микропроцессорное устройство, ориентированное на задачи циклового и программно-логического управления технологическим оборудованием. Алгоритм работы данного устройства определяется программой, вводимой в его память.

МПЦУ имеет модульный принцип построения. В его состав входят следующие блоки:

- 1) модуль преобразователя (МПРБ);
- 2) модуль стабилизатора 5В;
- 3) модуль стабилизатора 12В;
- 4) модуль выходов (Мвых);
- 5) модуль входов (Мвх);
- 6) модуль энергонезависимой памяти (МЭНП);
- 7) модуль памяти (МП);
- 8) модуль процессора (МПР);
- 9) пульт управления (ПУ).

В операциях обмена данными управляющим устройством МПЦУ является модуль процессора, исполнительным устройством - модули систем памяти и ввода-вывода.

Модуль процессора (МПР), выполненный на базе БИС КР580ИК80, осуществляет сбор, цифровую обработку и вывод информации в соответствии с исполнительной программой, записанной в перепрограммируемое постоянное запоминающее устройство (ПЗУ) модуля памяти.

Последовательностью операции обмена данными между модулем процессора и модулями-исполнителями управляет исполнительная программа, назначение которой состоит в преобразовании инструкций, введенных оператором с пульта управления или поступающих от управляющей программы, в последовательности кодов машинного языка микропроцессора, реализующих эти инструкции.

Управляющая программа - программа, написанная в кодах команд входного языка МПЦУ и обеспечивающая выполнение заданного алгоритма управления внешним оборудованием. Она размещается в модулях энергонезависимого ЗУ (МЭНЗУ).

Пульт управления (ПУ) совместно с модулем управления (МУ) составляют технические средства общения оператора с



МПЦУ. ПУ включает в себя клавиатуру для ввода команд и управления режимом работы МПЦУ, однострочный дисплей, предназначенный для отображения контролируемой оператором информации, и индикаторы режимов работы. МУ обеспечивает сопряжение клавиатуры и индикации ПУ с внутренней магистралью обмена информацией МПЦУ.

Модули ввода (МВВ) и вывода (МВЫВ) дискретных сигналов предназначены для связи МПЦУ с внешним технологическим оборудованием. Они обеспечивают преобразование уровней и гальваническую развязку сигналов, а также индикацию состояния каждого входа и выхода МПЦУ.

Название работы: Программирование промышленных роботов с использованием компьютерного эмулятора микропроцессорного циклового устройства (МПЦУ).

Цель работы: ознакомление с устройством, принципами работы и системой команд МПЦУ, получение навыков его программирования.

Режимы работы МПЦУ

Режимы работы МПЦУ представлены в таблице 1.

Переключение режимов происходит следующим образом: нажимается клавиша "P", затем номер режима (см. таблицу). Текущий режим работы МПЦУ показывается соответствующим индикатором режима на пульте управления. После загрузки система переходит в режим "Ручного управления".

Режимы работы МПЦУ

Номер режима	Название режима	Включенный индикатор
0	Автоматическая работа	А
1	Ручное управление	Р
2	Пошаговая работа	Ш
3	Ввод программы	ВП
4	Просмотр программы вперед	ПП
5	Просмотр программы назад	ПП



Робототехника и мехатроника

6	Функциональная клавиатура (робот 1)	P
7	Функциональная клавиатура (робот 2)	P

Режим №0 "Автоматическая работа"

Автоматический режим работы является основным и предназначен для управления технологическим оборудованием в соответствии с алгоритмом, реализованным в виде управляющей программы, хранимой в ЭНЗУ.

Режим №1 "Ручное управление"

В режиме ручного управления процессор взаимодействует с ПУ аналогично режиму "Ввод программы", однако сформированный код команды не записывается в ЭНЗУ, а выполняется. Содержимое счетчика команд при этом не изменяется и не индицируется.

Режим №2 "Пошаговая работа"

Данный режим является средством отладки управляющих программ. При этом режиме процессор осуществляет выборку из ОЗУ и выполнение очередной команды, после чего модифицирует содержимое счетчика команд и переходит в режим ожидания ввода с клавиатуры. Каждое нажатие клавиши, если оно не связано с изменением режима работы, переводит процессор на выполнение очередного шага программы.

Режим №3 "Ввод программы"

Используется для записи кодов команд управляющей программы в ЭНЗУ. Необходимая команда набирается оператором на клавиатуре ПУ. Процессор под управлением исполнительной команды считывает коды нажатых клавиш, формирует из них код команды и пересылает его в ЭНЗУ по адресу, определяемому счетчиком команд, который размещен в ОЗУ МП.

По окончании пересылки каждой команды содержимое счетчика команд увеличивается на единицу. Вводимая информация совместно с текущим значением счетчика команд отображается на дисплее ПУ.

Режимы №4 и №5 "Просмотр программы"

Используются для контроля оператором управляющей программ, хранящейся в памяти МПЦУ. В соответствии с адресом



ячейки ЭНЗУ, записанном в счетчике команд, производится считывание содержащейся в этой ячейке информации и вывод ее на индикацию совместно с текущим значением счетчика команд. Содержимое счетчика команд модифицируется в зависимости от режима (прибавляется 1 в режиме №4 или отнимается в режиме №5).

Режимы №6 и №7 "Функциональная клавиатура"

Используются для управления манипуляторами от клавиатуры с использованием подпрограмм, записанных в ПЗУ, обеспечивающих включение звеньев манипулятора нажатием одной клавиши. Код включения движения звена манипулятора вводится нажатием оператором соответствующей клавиши. Процессор под управлением исполнительной программы считывает код нажатой клавиши, отыскивает в ПЗУ соответствующую подпрограмму и приступает к ее выполнению. Содержание счетчика команд при этом не изменяется и не индицируется.

Регистры МПЦУ

"БА" - номер зоны памяти, из которой выбираются команды управляющей программы;

"СК" - счетчик команд, содержимое которого определяет адрес команды;

"БУ" - бит условия, значение которого устанавливается в зависимости от результата выполнения ряда операций. Анализ бита условия используется для управления кодом выполнения управляющей программы;

"УС" - регистр указатель стека, определяющий адреса ячеек ОЗУ, в который запоминаются значения "БА" и "СК" при обращении к подпрограмме, и откуда они выбираются при выполнении команды "Возврат";

шестнадцать счетчиков, содержание которых может изменяться от 0 до F. При необходимости счетчик может быть использован как однобитная память.

Система команд МПЦУ

Основные команды МПЦУ (см. таблицу 2) по функциональному назначению делятся на следующие группы:

1) команды ввода-вывода;



- 2) команды управления программой;
- 3) команды управления счетчиками;
- 4) команды контроля и редактирования программы;
- 5) команды тестового контроля функциональных блоков;
- 6) команды, ориентированные на управление пневматическими манипуляторами по путевому признаку.

Формат команды МПЦУ

Слово команды МПЦУ делится на два поля длиной по восемь разрядов: поле кода операции и поле операнда.

N0, N1, N2, N3 - кодируются символами шестнадцатеричной системы исчисления.

Таблица 2
Основные команды МПЦУ

Формат команды		Содержание команды
Код операции (N3 и N2)	Операнд (N1 и N0)	
<u>Команды ввода-вывода</u>		
01	адрес входа	ожидание отсутствия входного сигнала. Переход к выполнению следующей команды программы происходит только при отсутствии сигнала на входе с заданным адресом
02	адрес входа	ожидание наличия входного сигнала. Переход к выполнению следующей команды программы происходит только при наличии сигнала на входе с заданным адресом
03	адрес входа	проверка входа на отсутствие сигнала. При отсутствии сигнала на входе с заданным адресом бит условия сохраняет предыдущее значение, в противном случае бит условия обнуляется



Робототехника и мехатроника

04	адрес входа	проверка входа на наличие сигнала. При наличии сигнала на входе с заданным адресом бит условия сохраняет предыдущее значение, в противном случае бит условия обнуляется
05	адрес выхода	включить выход с заданным адресом
06	адрес выхода	выключить выход с заданным адресом
07	T	выдержка времени (команда задержки выполнения управляющей программы на время $T \cdot 0,1c$)
12	адрес выхода	вывод бита условия прямой (выход с заданным адресом устанавливается в соответствии с содержанием бита условия)
13	адрес выхода	вывод бита условия инверсный (выход с заданным адресом устанавливается в соответствии со значением, противоположным содержимому бита условия)
1E	адрес входа	бит условия принимает значение сигнала на входе с заданным адресом
1F	адрес входа	бит условия принимает инверсное значение сигнала на входе с заданным адресом
Команды управления программой		
00	00	нет операции (безусловный переход к выполнению следующей команды программы)
08	00	останов программы
09	адрес команды	безусловный переход к выполнению команды, содержащейся по указанному адресу
0A	адрес команды	переход к выполнению команды, содержащейся по указанному адресу, если в бите условия "1", в противном случае происходит выполнение следующей команды программы



Робототехника и мехатроника

0B	адрес команды	переход к выполнению команды, содержащейся по указанному адресу, если в бите условия "0", в противном случае происходит выполнение следующей команды программы
10	адрес команды	переход к подпрограмме (переход к выполнению команды, содержащейся по указанному адресу, с запоминанием адреса возврата)
11	00	возврат из подпрограммы
<u>Команды управления счетчиками</u>		
0D	0, номер счетчика	сброс счетчика (обнуление счетчика с заданным номером)
0C	0, номер счетчика	содержимое счетчика с заданным номером увеличивается на 1
0E	число, номер счетчика	сравнение счетчика (если содержимое счетчика с заданным номером равно числу, то бит условия сохраняет свое предыдущее значение, в противном случае бит условия обнуляется)
<u>Команды управления однобитовой памятью</u>		
20	адрес бита	установка адресованного бита в "1"
21	адрес бита	установка адресованного бита в "0"
22	адрес бита	считывание содержимого адресованного бита и сравнение с "1" (если содержимое адресованного бита соответствует "1" бит условия сохраняет своё значение, в противном случае бит условия обнуляется)



23	адрес бита	считывание содержимого адресованного бита и сравнение с "0" (если содержимое адресованного бита соответствует "0", бит условия сохраняет своё значение, в противном случае бит условия обнуляется)
24	адрес бита	установка адресованного бита в соответствии со значением бита условия

Таблица 3
Команды режима функциональной клавиатуры

Клавиша	Команда	Операнд	Движение	Включение выхода
0	F0	00 (01)	выдвижение руки	0
1	F1	00 (01)	втягивание руки	1
2	F3	00 (01)	поворот вправо	3
3	F2	00 (01)	поворот влево	2
4	F4	00 (01)	подъем	4
5	F5	00 (01)	опускание	5
A	FA	00 (01)	разжим схвата	A
B	FB	00 (01)	зажим схвата	–



ПРОГРАММИРОВАНИЕ МПЦУ В РЕЖИМЕ ЭМУЛЯЦИИ

Компьютерный эмулятор использует максимально приближенный интерфейс и возможности данного МПЦУ, что позволяет тренировать навыки в программировании и работы с реальным устройством. Для легкого понимания основ программирования и особенностей построения МПЦУ в тренажер встроена функция для вывода информации о состоянии внутренних регистров после выполнения каждой команды, также тренажер показывает состояние входных и выходных битов в процессе выполнения команд. В эмуляторе к МПЦУ подключены два цикловых робота, которые изменяют свое положение в зависимости от работы управляющей программы (в автоматическом режиме) или от действий оператора (в ручном режиме).

В компьютерном эмуляторе состояние входных и выходных сигналов отображается в соответствующих модулях (три модуля ввода и вывода). К модулям с номерами 0 и 1 подключены два манипулятора, третий модуль ввода (номер 2) используется для подачи входных сигналов от возможного оборудования (состояние входов для модуля ввода 2 определяется наличием отметки в нужном бите; для задания сигнала необходимо кликнуть на нужном бите).

Если во время работы необходимо восстановить исходное положение или остановить выполнение какой-нибудь команды или программы, то нужно нажать на клавишу сброс. При этом регистры обнуляются, манипуляторы принимают начальные положения, а введенная программа остается без изменения.

После набора программы в эмуляторе встроена возможность сохранения введенной программы и последующего ее считывания.

Чтобы ранее набранная программа не мешала программированию, в эмулятор введена процедура очистки памяти.



Задание на вариант №1

Алгоритм работы РТК:

Выставление звеньев роботов 0 и 1 в исходное положение (см. рис. ниже).

Если станок свободен (появление сигнала на входе 20 модуля ввода 2), то робот 0 берет из накопителя заготовку и подает ее на станок. Для этого роботу 0 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться вправо, опуститься, вытянуть "руку", задержка времени 5 сек, разжать схват, втянуть "руку".

При окончании обработки заготовки на станке (появление сигнала на входе 21 модуля ввода 2) робот 1 берет из станка деталь и подает ее на конвейер. Для этого роботу 1 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться вправо, опуститься, вытянуть "руку", разжать схват, втянуть "руку".

При появлении сигнала на входе 22 модуля ввода 2 повторить действия, начиная с пункта 1 данного алгоритма работы.

Исходное положение роботов

The screenshot shows the 'Модель МПЦУ' (MFCU Model) interface. It features a central diagram of two robots, 'МВЫВ0' and 'МВЫВ1', in their initial positions. The interface includes a menu bar with 'Операции с памятью', 'Помощь', and 'Выход'. On the right, there are two vertical columns of indicator lights labeled 'МВЫВ2' and 'МВВ 2', each with lights for digits 0-9 and letters A-F. Below the diagram is a control panel with a grid of indicator lights for addresses 0-7 and letters A-F, and buttons for 'ОЖ', 'СБР', 'А', 'Р', 'Ш', 'БП', 'ПП', 'C', 'D', 'E', 'F', '8', '9', 'A', 'B', '4', '5', '6', '7', 'P', '0', '1', '2', '3'. On the far right, a 'Значение регистров' (Register Values) section displays various status indicators like 'Блок адреса', 'Счетчик команд', 'Бит условия', 'Указатель стека', and 'Счетчики'.



Задание на вариант №2

Алгоритм работы РТК:

Выставление звеньев роботов 0 и 1 в исходное положение (см. рис. ниже).

Если станок свободен (появление сигнала на входе 23 модуля ввода 2), то робот 1 берет из накопителя заготовку и подает ее на станок. Для этого роботу 1 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться влево, опуститься, вытянуть "руку", задержка времени 10 сек, разжать схват, втянуть "руку".

При окончании обработки заготовки на станке (появление сигнала на входе 24 модуля ввода 2) робот 0 берет из станка деталь и подает ее на конвейер. Для этого роботу 0 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться влево, опуститься, вытянуть "руку", разжать схват, втянуть "руку".

При появлении сигнала на входе 25 модуля ввода 2 повторить действия, начиная с пункта 1 данного алгоритма работы.

Исходное положение роботов

The screenshot shows the 'Модель МПЦУ' (MPCU Model) software interface. The main window displays two robots in their initial positions. The interface includes a menu bar with 'Операции с памятью', 'Помощь', and 'Выход'. Below the main window is a control panel with various indicators and controls.

Исходное положение роботов

Модель МПЦУ

Операции с памятью Помощь Выход

МБВВ2 МБВ 2

0 0 1 1
 1 1 2 2
 2 2 3 3
 3 3 4 4
 4 4 5 5
 5 5 6 6
 6 6 7 7
 7 7 8 8
 8 8 9 9
 A A A A
 B B B B
 C C C C
 D D D D
 E E E E
 F F F F

0 0 8 8 0 0 8 8 0 0 8 8 0 0 8 8
 1 1 9 9 1 1 9 9 1 1 9 9 1 1 9 9
 2 2 A A 2 2 A A 2 2 A A 2 2 A A
 3 3 B B 3 3 B B 3 3 B B 3 3 B B
 4 4 C C 4 4 C C 4 4 C C 4 4 C C
 5 5 D D 5 5 D D 5 5 D D 5 5 D D
 6 6 E E 6 6 E E 6 6 E E 6 6 E E
 7 7 F F 7 7 F F 7 7 F F 7 7 F F

МБВВ0 МБВВ0 МБВВ1 МБВВ1

ОЖ

СБР

Адрес Код Операнд

А Р Ш ВП ПП

С D E F
 8 9 A B
 4 5 6 7
 P 0 1 2 3

Значение регистров

Блок адреса: 0
 Счетчик команд: 0
 Бит условия: 1
 Указатель стека: 0
 Счетчики:

0: 0 8: 0
 1: 0 9: 0
 2: 0 A: 0
 3: 0 B: 0
 4: 0 C: 0
 5: 0 D: 0
 6: 0 E: 0
 7: 0 F: 0



Задание на вариант №3

Алгоритм работы РТК:

Выставление звеньев роботов 0 и 1 в исходное положение (см. рис. ниже).

Если в накопителе N1 имеется заготовка (появление сигнала на входе 26 модуля ввода 2), робот 0 берет из N1 заготовку и подает ее на станок. Для этого роботу 0 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", опуститься, развернуться вправо, вытянуть "руку", задержка времени 3 сек, разжать схват, втянуть "руку".

Если в накопителе N2 имеется заготовка (появление сигнала на входе 27 модуля ввода 2) робот 1 берет из N2 заготовку и подает ее на станок. Для этого роботу 1 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", опуститься, развернуться влево, вытянуть "руку", задержка времени 3 сек, разжать схват, втянуть "руку".

При появлении сигнала на входе 28 модуля ввода повторить действия, начиная с пункта 1 данного алгоритма работы.

Исходное положение роботов

Блок адреса: 0	
Счетчик команд: 0	
Бит условия: 1	
Указатель стека: 0	
Счетчики:	
0: 0	8: 0
1: 0	9: 0
2: 0	A: 0
3: 0	B: 0
4: 0	C: 0
5: 0	D: 0
6: 0	E: 0
7: 0	F: 0



Задание на вариант №4

Алгоритм работы РТК:

Выставление звеньев роботов 0 и 1 в исходное положение (см. рис. ниже).

Если станок С1 свободен (появление сигнала на входе 29 модуля ввода 2), то робот 0 берет из накопителя заготовку и подает ее на С1. Для этого роботу 0 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться влево, опуститься, вытянуть "руку", задержка времени 6 сек, разжать схват, втянуть "руку".

Если станок С2 свободен (появление сигнала на входе 2А модуля ввода 2) робот 1 берет из накопителя заготовку и подает ее на С2. Для этого роботу 1 необходимо выполнить следующие действия: вытянуть "руку", сжать схват, втянуть "руку", подняться, развернуться вправо, опуститься, вытянуть "руку", задержка времени 8 сек, разжать схват, втянуть "руку".

При появлении сигнала на входе 2В модуля ввода повторить действия, начиная с пункта 1 данного алгоритма работы.

Исходное положение роботов

The screenshot shows the 'Модель МПЦУ' (MFCU Model) software interface. It features a central diagram of two robot arms, each with a green base and a blue arm. To the right of the diagram are two vertical columns of indicator lights labeled 'МВЫВ2' and 'МВВ 2', with letters A through F corresponding to the lights. Below the diagram is a control panel with a grid of indicator lights for inputs 0-9 and A-F, and outputs 0-9 and A-F. The panel also includes a 'Значение регистров' (Register Values) section with fields for 'Блок адреса', 'Счетчик команд', 'Бит условия', and 'Указатель стека'. A numeric keypad and function buttons (A, P, Ш, ВП, ПП) are also visible.

Отчет по лабораторной работе должен содержать:



Робототехника и мехатроника

Фамилию студента, название группы, дату сдачи.

Название работы, ее цель.

Задание, соответствующее конкретному варианту (исходное положение робота, алгоритм его работы).

Схему компоновки РТК.

Текст программы, соответствующей конкретному варианту (требование к программе: выставление роботов в исходное положение необходимо проводить посредством проверки состояния выходов на модулях вывода 0 и 1).

Контрольные вопросы ориентированы на проверку знаний студента в области:

- 1) общих сведений об устройстве и принципах работы МПЦУ;
- 2) используемых в МПЦУ основных командах: формат и их краткое содержание;
- 3) порядка программирования МПЦУ.



ОПИСАНИЕ СИСТЕМЫ ПРОГРАММИРОВАНИЯ SRCL

Порядок выполнения работы

1. В отчете нарисовать схему компоновки РТК, в состав которого, помимо ПР KUKA IR 161/15, входит различное технологическое оборудование (металлообрабатывающие станки, бункерно-загрузочные устройства для заготовок или готовых изделий, конвейерные линии и т.д.).

2. Написать алгоритм работы робота для созданного РТК.

3. Реализовать полученный алгоритм работы робота посредством программирования с помощью компьютерного эмулятора.

Для начала работы с эмулятором необходимо запустить файл Car.exe и открыть дополнительные окна, в которых изображен пульт управления и дисплей. При необходимости изображение робота можно повернуть стрелками на клавиатуре и увеличить (уменьшить) клавишами INS (Delete).

Программирование робота осуществляется в следующей последовательности:

Включить напряжение сети.

Включить главный выключатель.

Включить систему управления робота.

Включить приводы.

Отжать кнопку "Программирование".

Отжать кнопку "Доп.ввод".

Отжать кнопку "DEF".

Ввести двухзначный номер новой программы от 01 до 99.

Отжать кнопку "Ввод".

Отжать кнопку "GES" и задать трехзначное значение скорости BAN.

Отжать кнопку "Ввод".

Отжать кнопку "GES" и кнопкой "Выбор" выбрать ALL.

Задать трехзначное значение скорости ALL (рекомендация: то же значение, что и у скорости BAN).

Отжать кнопку "Ввод".

Отжать кнопку "РТР".

Кнопками управления по осям, вывести робот в точку.

При достижении нужной точки запомнить ее, отжав кнопку "РТР".

Повторять пункты 15-16 нужное число раз.



Программирование завершается при отжати кнопки "ручн" (при этом программа завершается сообщением "END HPxx").

После завершения программирования создается файл HPxx.dat. После программирования для загрузки программы требуется перезапуск эмулятора.

Загрузка программы требует выполнения следующих пунктов:

Выполнить пункты 1-4 руководства к программированию.

Отжать кнопку "ANW".

Ввести номер созданной программы.

Отжать "ANW".

Отжать кнопку "Ввод".

Отжать кнопку "Цикл" на пульте "Автоматика".

Отжать кнопку "+Пуск" (под аварийным выключателем).

Отчет по лабораторной работе должен содержать:

Фамилию студента, название группы, дату сдачи.

Название работы, ее цель.

Схему компоновки РТК.

Алгоритм работы робота.

Текст программы.

Координаты точек, используемых программой.

Контрольные вопросы ориентированы на проверку знаний студента в области:

- 1) общих сведений о языке программирования SRCL;
- 2) используемых в данном языке команд: их назначение, формат;
- 3) порядка программирования ПР KUKA RC20/41.