

УДК 681.5

Магистрант М.В. Клютко

Науч. рук. доц. Д.С. Карпович

(кафедра автоматизации производственных процессов и электротехники, БГТУ)

ОСОБЕННОСТИ AS – ПРОГРАММИРОВАНИЯ ПРОМЫШЛЕННОГО МАНИПУЛЯТОРА KAWASAKI FS 03N

В AS (Automation Solution) системе можно задавать команды или выполнять программы, используя AS язык программирования. AS система записана в энергонезависимой памяти контроллера (ОЗУ). Когда управляющее питание включается, AS система стартует и ожидает ввода команд для выполнения.

AS система управляет роботом в соответствии с данными командами и программами. Во время выполнения программы могут использоваться некоторые типы функций такие как, отображение статус состояния системы, позиционное текущее положение робота, запись данных во внешние устройства памяти, написание и редактирование программ.

В AS системе роботы управляются и функционируют, основываясь на программе, которая создается до выполнения действий и описывает последовательность действий, необходимых для решения поставленной задачи.

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 используются внутри программы, как программные инструкции.

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

В AS системе возможны два типа задания координатной точки в трехмерном пространстве: последовательностью угловых смещений каждой из осей суставов, относительно какого-то начального положения (угловая координата), последовательностью линейных и угловых смещений начала базовой системы координат (декартова координата).

1. Угловая координата

Используя значения кодера (кодирующего устройства), по количеству оборотов вала каждого из суставов рассчитываются угловые смещения каждой из осей суставов относительно какого-то начального

положения. Последовательность величин смещений каждой оси задается в градусах. Данная последовательность при запоминании задаст положение и ориентацию центра фланца робота в пространстве. При запоминании точки в угловых координатах необходимо перед буквой поставить знак #.

```

JT1   JT2   JT3   JT4   JT5   JT6
#pose= 0.00, 33.00, -15.00, 0, -40, 30
    
```

2. Декартова координата

Описывает положение координат по отношению с рекомендуемыми координатами. Если не определено иначе, это относится к значениям преобразования координат инструмента относительно основных координат робота. Позиция дается значениями XYZ для основных координат, и положением O, A, T углов Эйлера.

Положение и ориентация центра фланца в пространстве задается последовательностью смещений начала инструментальной системы координат относительно начала базовой системы координат координатами X, Y, Z, и углами Эйлера O, A, T.

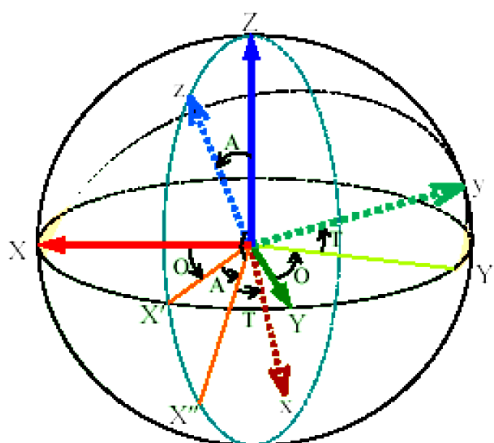


Рисунок 1 – Углы Эйлера

O – угол между плоскостью Zz и плоскостью XZ

A – угол между осью z и осью Z

T – угол между осью x и осью X''

Ось X'' лежит в плоскости Zz и образует с осью z угол 90°.

```

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

Программирование осуществляется, как показано на рисунке 2.

Рассмотрели использование относительных позиций (рисунке 3).

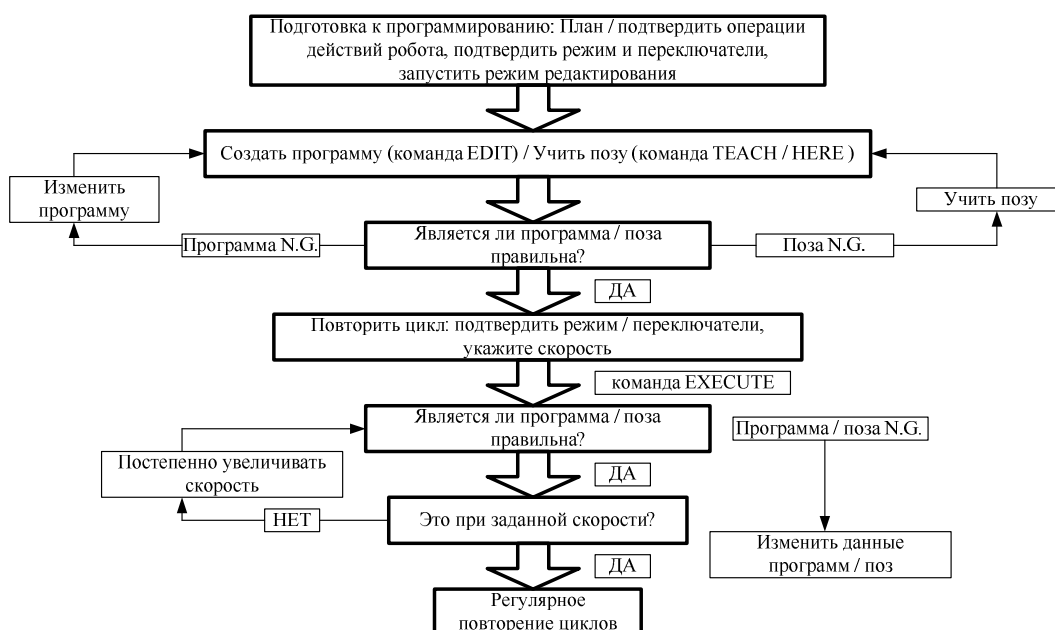


Рисунок 2 – Процедура программирования промышленного робота

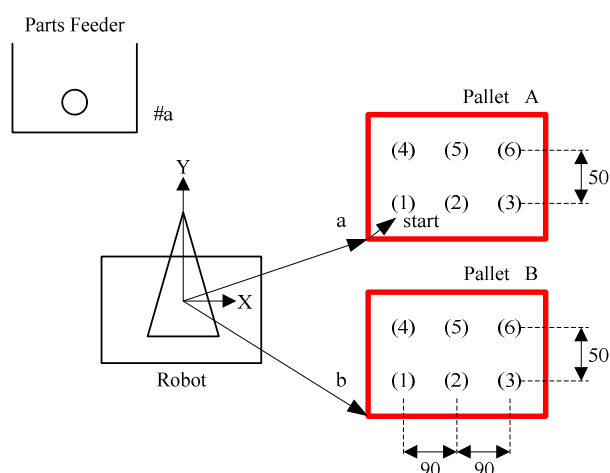


Рисунок 3 – Размещение робота, поддонов и подающего лотка

Детали берутся с подающего лотка и раскладываются на поддоны в порядке от (1) до (6) в два ряда (на расстоянии 50 мм) и три столбца (на расстоянии 90 мм). Для упрощения и поддоны и детали, помещаемые на поддон, установлены параллельно базовой системе координат робота. Поддоны помещены отдельно, но зависимость между контрольной точкой и местами, где детали должны быть размещены, одинаковы на любом из поддонов. Эта программа выполняет установку детали на поддон А. После того, как шесть деталей установлены, робот продолжает делать то же самое, устанавливая на поддон В.

Позиции для обучения: #a: позиция, где робот берет детали с подающего лотка; a: контрольная точка на поддоне А; b: контрольная точка на поддоне В; start: позиция первой детали на поддоне.

```
Программа:.PROGRAM relative.test
;начальные установки (2 ряда, 3 столбца, ΔX=90, ΔY=50, etc.)
row.max=2; col.max=3; xs=90; ys=50
OPENI
flg=0; flg=0: Pallet A, flg=1: Pallet B
POINT pallet=a; начало пакетирования деталей
10 POINT put=start
FOR row=1 TO row.max
FOR col=1 TO col.max
JAPPRO #a, 100
LMOVE #a; берет деталь с подающего лотка
CLOSEI
LDEPART 100;
POINT put_pt=pallet+put
JAPPRO put pt, 200
LMOVE ; кладет деталь на поддон
OPENI
LDEPART 200;
POINT put=SHIFT(put BY xs,0,0) ; определяет место для детали в
следующем столбце
END;
POINT put=SHIFT(put BY 0,ys*row,0) ; определяет место для дета-
ли в следующей строке
END;
IF flg<>0 GOTO 30 ; переходит к финишной процедуре (flg=1)
flg=1
POINT pallet=b ; определяет контрольную точку на поддоне В
GOTO 10
30 TYPE "***end***"
STOP
.END
```

Были выявлены достоинства AS системы: робот может двигаться непрерывно вдоль заданной траектории и при обучении позициям или при повторении действий робот может идти по линейной траектории, сохраняя положение инструмента неизменным.

На основании AS – языка программирования была получена программа перемещения с подающего лотка на два поддона с использованием относительных позиций.