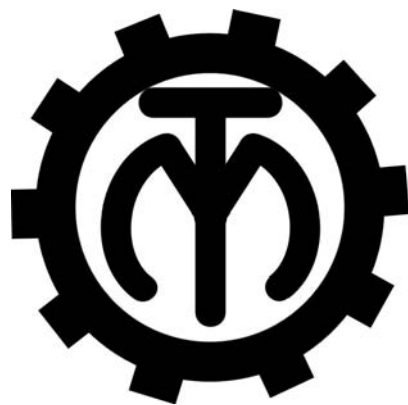


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Технология машиностроения»

# ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ С НЕЧЕТКОЙ ЛОГИКОЙ В МЕХАТРОНИКЕ И РОБОТОТЕХНИКЕ

*Методические рекомендации к лабораторным работам  
для магистрантов направления подготовки  
15.04.06 «Мехатроника и робототехника»  
очной и заочной форм обучения*



Могилев 2023

УДК 621.865.8  
ББК 32.816  
И86

Рекомендовано к изданию  
учебно-методическим отделом  
Белорусско-Российского университета

Одобрено кафедрой «Технология машиностроения» «22» ноября 2022 г.,  
протокол № 6

Составитель д-р техн. наук, проф. В. М. Пашкевич

Рецензент канд. техн. наук, доц. А. П. Прудников

В методических рекомендациях даны задания к лабораторным работам по дисциплине «Искусственный интеллект с нечеткой логикой в мехатронике и робототехнике», приведены указания по их выполнению.

Учебно-методическое издание

## ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ С НЕЧЕТКОЙ ЛОГИКОЙ В МЕХАТРОНИКЕ И РОБОТОТЕХНИКЕ

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 36 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, 2023

## Содержание

1 Построение модели и алгоритма управления мобильного робота.....	4
2 Построение модели и алгоритма управления робота-пылесоса.....	7
3 Построение модели и алгоритма управления робота-сортировщика.....	11
4 Построение модели и алгоритма управления робота-сборщика.....	14
Список литературы.....	19

# 1 Построение модели и алгоритма управления мобильного робота

## Задание 1

По вариантам, выданным преподавателем, построить функции принадлежности нечетких переменных для системы управления скоростью перемещения робота  $V$ , м/с, в зависимости от расстояния до препятствия  $L$ , м, для набора правил:

- 1) «Если расстояние до препятствия большое, то скорость высокая»;
- 2) «Если расстояние до препятствия среднее, то скорость средняя»;
- 3) «Если расстояние до препятствия малое, то скорость низкая».

## *Порядок выполнения работы*

Построить графики дискретных функций принадлежности. В качестве функции принадлежности принять треугольную или трапециевидную функцию в форме выражений вида

$$\mu(M) = \frac{M - M_{ep}}{\Delta M}$$

или

$$\mu(M) = \frac{M_{ep} - M}{\Delta M},$$

где  $M$  – текущее значение момента, Н·м;

$M_{ep}$  – значение момента, соответствующее началу (концу) переходной зоны, Н·м;

$\Delta M$  – ширина переходной зоны, Н·м.

Интервал расстояний – 0,1...4,1 м; интервал скоростей – 0,2...8,2 м/с.

Пример рабочего листа приведен на рисунке 1.

## *Вопросы к защите*

- 1 Чем отличается нечеткое множество от обычного?
- 2 Какие функции нечеткости используют на практике?
- 3 Что такое функция принадлежности?

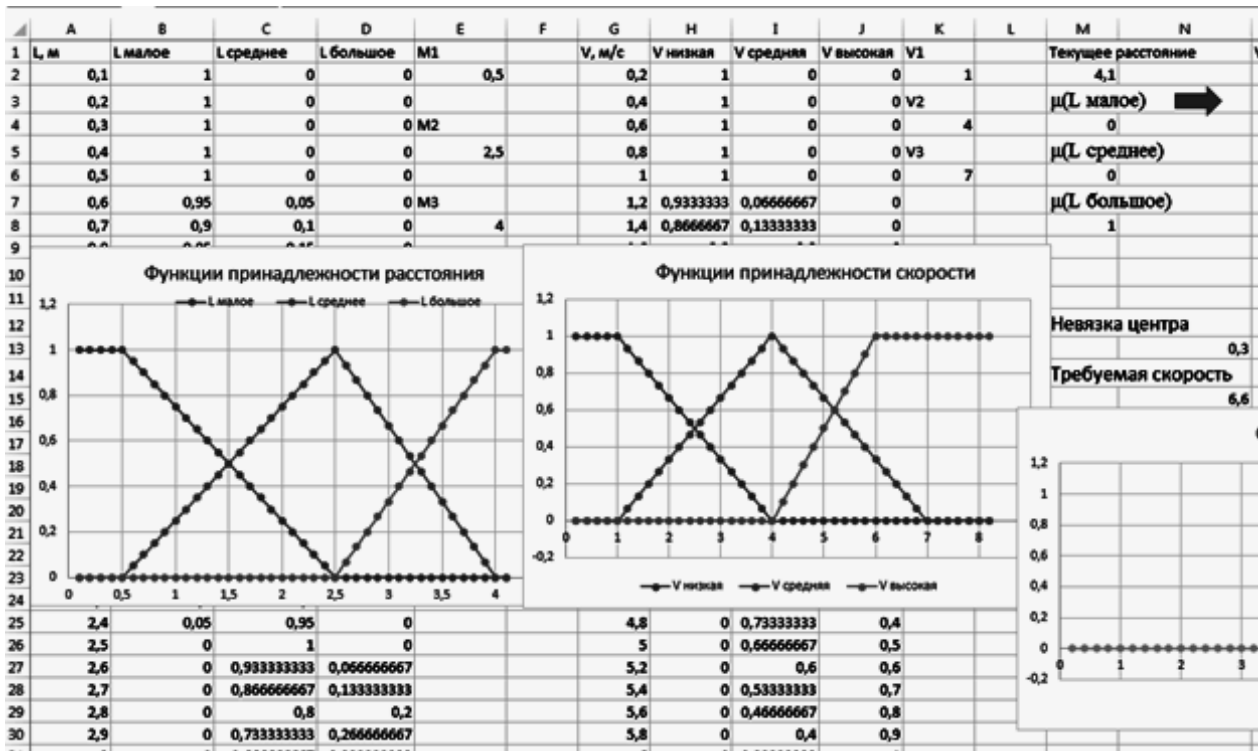


Рисунок 1 – Пример оформления отчета

## Задание 2

Используя функции принадлежности, построенные ранее, определить функцию управления скоростью перемещения по величине расстояния по методу центра тяжести (Мамдани).

### Порядок выполнения работы

Для каждого дискретного значения расстояния поочередно необходимо выполнить следующее.

1 Определить истинность левой части правил («Если») – для множеств «Расстояние малое», «Расстояние среднее», «Расстояние высокое».

Для поиска позиции и содержимого соответствующих ячеек рекомендуется использовать функции ПОИСКПОЗ и ИНДЕКС.

2 Рассчитать функции принадлежности для правой части правил («То») – для множеств «Скорость низкая», «Скорость средняя», «Скорость высокая»:

$$\mu_{TO}(x) = \mu_{ЕСЛИ}(x) \cdot \mu_{ПРАВИЛО}(x).$$

3 Определить композицию (суперпозицию) нечетких множеств методом «Или»

$$\mu_{\Sigma}(x) = \max \{ \mu_{TO}(x) \}.$$

4 Провести дефаззификацию (переход к единственному значению) переменной по методу Мамдани (на основе определения центра тяжести).

С этой целью рассчитать накопленные суммы функций принадлежности от начала до каждой текущей позиции в столбце суперпозиций (сумма слева) и от конца до текущей позиции (сумма справа).

Найти модуль разницы сумм слева и справа (столбец «Разность»).

Найти минимальную разницу сумм слева и справа, приближенно соответствующую центру тяжести суперпозиции нечетких множеств. Определить индекс соответствующей ячейки и требуемую скорость перемещения.

5 Построить график управления скоростью перемещения по величине расстояния до заготовки.

Пример рабочего листа с отчетом приведен на рисунке 2.

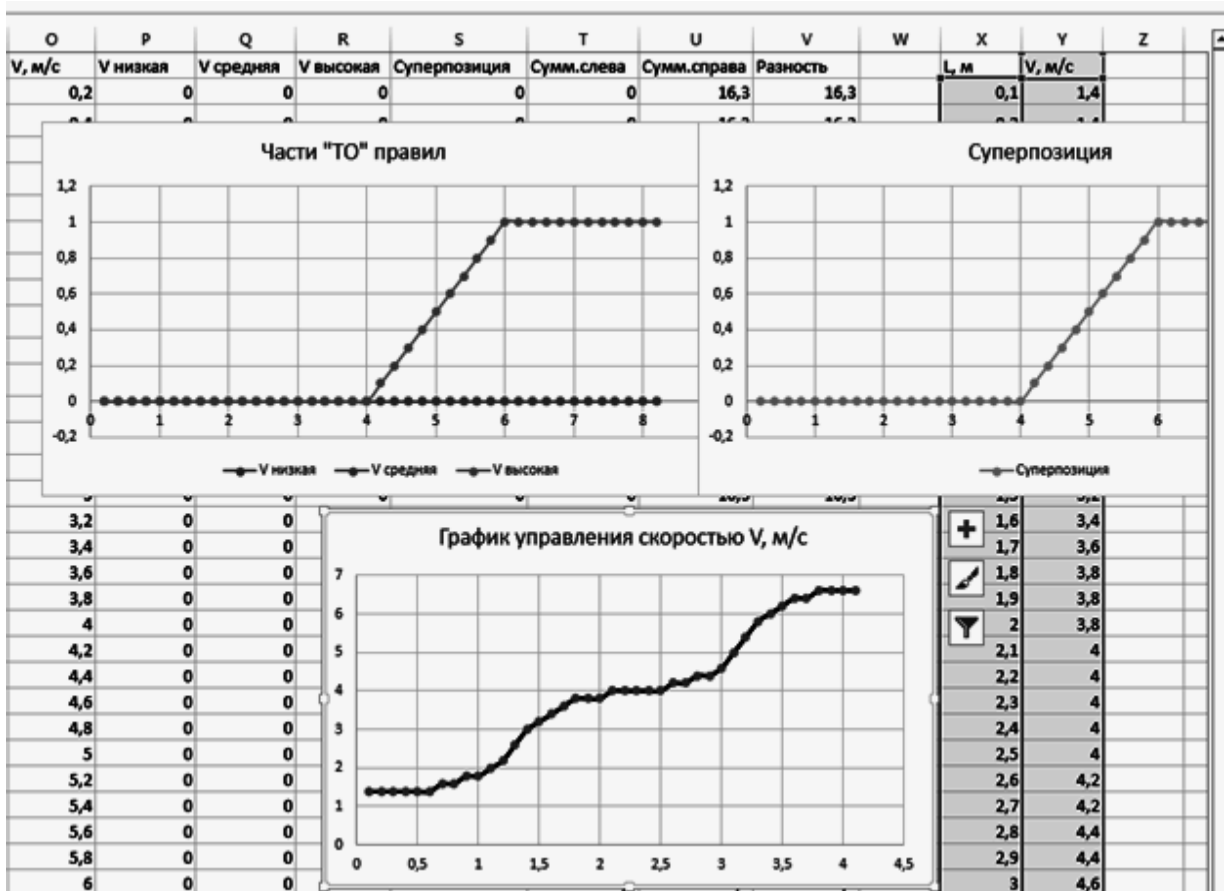


Рисунок 2 – Пример оформления отчета

### **Вопросы к защите**

- 1 Что такое композиция в нечетком выводе?
- 2 Что такое дефаззификация?
- 3 В чем сущность дефаззификации по методу Мамдани?

## 2 Построение модели и алгоритма управления работа-пылесоса

### Задание

По заданию, выданному преподавателем, исследовать пространство поиска работа-пылесоса.

Рассмотреть различные варианты перемещений с произвольным исходным положением робота.

Исследовать влияние препятствий на результаты работы робота.

### Порядок выполнения работы

Установить начальное положение робота (строка / столбец рабочего поля). Рабочее пространство размером  $10 \times 10$  выделяется красным цветом; очищенная поверхность – желтым (рисунок 3).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1		1	2	3	4	5	6	7	8	9	10				Старт		1 строка
2	1																10 столбец
3	2																
4	3														Текущая		10 строка
5	4																4 столбец
6	5																
7	6														Шаг #		16
8	7																
9	8																
10	9																
11	10																
12																	
13																	
14																	
15																	
16																	
17																	
18																	
19																	

Рисунок 3 – Рабочее пространство робота

Перемещение робота управляется приведенным ниже макросом, включающим главную управляющую процедуру (Main); процедур Paint для подготовки поля к работе (закраски красным), Step для установки маркера положения робота в начальную точку и функций проверки возможности перемещения влево stepL, вправо stepR, вверх stepU, вниз stepD.

```
Sub Main()
```

```
    Paint ' Изобразить поле
    Start ' Установить робота в начальное поле
```

```
    ' Движение робота
    For i = 1 To 99
        Step
    Next i
```

```
End Sub
```

```
Sub Paint() ' Закраска начального поля
```

```
    For i = 1 To 10
        For j = 1 To 10
            Range("a1").Offset(i, j).Select
            Selection.Interior.Color = 255 ' Красный
        Next j
    Next i
```

```
End Sub
```

```
Sub Start() ' Установка в стартовую позицию
```

```
    i = Range("p1")
    j = Range("p2")
    Cells(i + 1, j + 1).Select
    Selection.Interior.Color = 65535 ' yellow

    Range("p4") = i ' Начальная позиция в строке
    Range("p5") = j ' Начальная позиция по столбцу

    Range("p7") = 1 ' Номер шага
```

```
End Sub
```

```
Sub Step() ' Управление перемещением робота
```

```
    ans = stepR()
    If ans = 1 Then ' Если шаг вправо возможен – переместиться
        i = Range("p4")
        j = Range("p5")
        Cells(i + 1, j + 2).Select
        Selection.Interior.Color = 65535
```



```

    Range("p4") = i
    Range("p5") = j + 1
    Range("p7") = Range("p7") + 1
    Exit Sub
End If

ans = stepD() ' Если шаг вниз возможен – переместиться
If ans = 1 Then
    i = Range("p4")
    j = Range("p5")
    Cells(i + 2, j + 1).Select
    Selection.Interior.Color = 65535
    Range("p4") = i + 1
    Range("p5") = j
    Range("p7") = Range("p7") + 1
    Exit Sub
End If

ans = stepL() ' Если шаг влево возможен – переместиться
If ans = 1 Then
    i = Range("p4")
    j = Range("p5")
    Cells(i + 1, j).Select
    Selection.Interior.Color = 65535
    Range("p4") = i
    Range("p5") = j - 1
    Range("p7") = Range("p7") + 1
    Exit Sub
End If

ans = stepU() ' Если шаг вверх возможен – переместиться
If ans = 1 Then
    i = Range("p4")
    j = Range("p5")
    Cells(i, j + 1).Select
    Selection.Interior.Color = 65535
    Range("p4") = i - 1
    Range("p5") = j
    Range("p7") = Range("p7") + 1
    Exit Sub
End If

End Sub

Function stepL() ' Проверка возможности шага влево

```

```

i = Range("p4") ' начальная позиция (i+1; j+1) строка/столбец
j = Range("p5")
Cells(i + 1, j).Select ' столбец: -1
If Selection.Interior.Color = 255 Then stepL = 1 Else stepL = 0
End Function

```

```

Function stepR() ' Проверка возможности шага вправо
i = Range("p4") ' начальная позиция (i+1; j+1) строка/столбец
j = Range("p5")
Cells(i + 1, j + 2).Select ' столбец: +1
If Selection.Interior.Color = 255 Then stepR = 1 Else stepR = 0
End Function

```

```

Function stepU() ' Проверка возможности шага вниз
i = Range("p4") ' начальная позиция (i+1; j+1) строка/столбец
j = Range("p5")
Cells(i, j + 1).Select ' строка: -1
If Selection.Interior.Color = 255 Then stepU = 1 Else stepU = 0
End Function

```

```

Function stepD() ' Проверка возможности шага вверх
i = Range("p4") ' начальная позиция (i+1; j+1) строка/столбец
j = Range("p5")
Cells(i + 2, j + 1).Select ' строка: +1
If Selection.Interior.Color = 255 Then stepD = 1 Else stepD = 0
End Function

```

Исследовать пространство поиска робота при наличии и отсутствии препятствий.

### ***Вопросы к защите***

1 Каким образом может быть построена траектория перемещения робота в свободном пространстве?

2 Каким образом может быть построена траектория перемещения робота при наличии препятствий?

3 Какие способы организации траектории перемещения можно предложить для автономного робота?

### 3 Построение модели и алгоритма управления работа-сортировщика

#### Задание

По заданию, выданному преподавателем, исследовать алгоритм работы работа-сортировщика.

На роторный конвейер в случайном порядке поступают три вида заготовок, маркированных красной, желтой и зеленой метками. Заготовки распознаются роботом и убираются в упаковки по  $N$  комплектов или в промежуточный накопитель для временного хранения.

Построить алгоритм сортировки для работы без накопителя и с накопителем.

Изучить влияние объема упаковки на производительность работа.

Изучить влияние количества параллельно заполняемых упаковок на производительность работа.

#### Порядок выполнения работы

Рабочее пространство работа содержит устанавливаемые параметры «Комплектов на упаковку», «Объем накопителя» (рисунок 4). Информационные параметры – «Упаковка сформирована», а также цветовая схема укомплектованных упаковок.

	A	B	C	D	E	F	G	H	I	J	K
1			Объем накопителя	0							
2											
3			Комплектов на упаковку	4							
4											
5			Готово упаковок	0							
6											
7			Упаковка №	1	2	3	4	5			
8											
9			Красных	2	4	0	0	0			
10			Желтых	2	4	0	0	0			
11			Зеленых	2	4	0	0	0			
12											
13			Упаковка сформирована	1	0	0	0	0			
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											

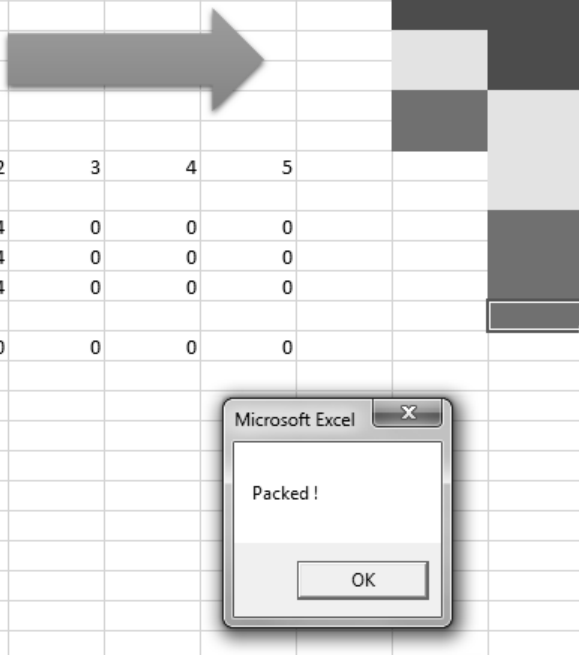


Рисунок 4 – Рабочее пространство работа

Управляющий макрос с интуитивно понятными именами переменных представлен ниже.

```
Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
```

```
' подключение библиотеки с таймером Windows
```

```
Sub Main()
```

```
UnitsInBox = Range("D3") ' количество заготовок одного цвета в
комплекте
```

```
For i = 1 To 5
```

```
ans = Range("c13").Offset(0, i)
```

```
If ans = 0 Then
```

```
NumUpak = i ' номер незаполненной упаковки
```

```
Exit For
```

```
End If
```

```
Next i
```

```
Clr = InsertObj ' на конвейере появляется заготовка.
```

```
' Распознается ее цвет Clr
```

```
Wait (500) ' ожидание 500 мс
```

```
' заполнена ли упаковка?
```

```
PackIsReady = Range("d13").Offset(0, NumUpak)
```

```
If PackIsReady = 0 Then ' если незаполнена:
```

```
RInPack = Range("c9").Offset(0, NumUpak)
```

```
' красных заготовок в упаковке
```

```
YInPack = Range("c10").Offset(0, NumUpak)
```

```
' желтых заготовок в упаковке
```

```
GInPack = Range("c11").Offset(0, NumUpak)
```

```
' зеленых заготовок в упаковке
```

```
End If
```

```
If Clr = 255 And RInPack < UnitsInBox Then
```

```
' если заготовка красная и комплект красных не собран
```

```
Range("a1").Select
```

```
Selection.Interior.Color = xlNone
```

```
' перенести заготовку в свободное место упаковки
```

```
Wait (100)
```

```
Range("i1").Offset(RInPack, NumUpak).Select
```

```
Selection.Interior.Color = Clr
```

```
Range("c9").Offset(0, NumUpak) = RInPack + 1
```

```
' установить счетчик красных заготовок на единицу больше
```

End If

If Clr = 65535 And YInPack < UnitsInBox Then

' если заготовка желтая и комплект желтых не собран

Range("a1").Select

Selection.Interior.Color = xlNone

Wait (100)

Range("i1").Offset(UnitsInBox + YInPack, NumUpak).Select

Selection.Interior.Color = Clr

Range("c10").Offset(0, NumUpak) = YInPack + 1

End If

If Clr = 5287936 And GInPack < UnitsInBox Then

' если заготовка зеленая и комплект зеленых не собран

Range("a1").Select

Selection.Interior.Color = xlNone

Wait (100)

Range("i1").Offset(UnitsInBox \* 2 + GInPack, NumUpak).Select

Selection.Interior.Color = Clr

Range("c11").Offset(0, NumUpak) = GInPack + 1

End If

' Проверить число заготовок каждого цвета в упаковке

RInPack = Range("c9").Offset(0, NumUpak)

YInPack = Range("c10").Offset(0, NumUpak)

GInPack = Range("c11").Offset(0, NumUpak)

If RInPack = UnitsInBox And YInPack = UnitsInBox And GInPack =  
UnitsInBox Then ' если упаковка заполнена

MsgBox ("Packed !")

' Установить счетчик заполненных упаковок на единицу больше

Range("c13").Offset(0, NumUpak) = 1

End If

End Sub

' процедура ожидания ms миллисекунд для улучшения визуализации

Sub Wait(ms)

Debug.Print Now()

Sleep ms

End Sub

```

Function InsertObj() ' выставить заготовку на конвейер и проверить ее цвет
lbl1: ans = Round(3 * Rnd())
      If ans = 0 Then GoTo lbl1
      If ans = 1 Then Clr = 255      ' Red
      If ans = 2 Then Clr = 65535   ' Yellow
      If ans = 3 Then Clr = 5287936 ' Green

      Range("a1").Select
      Selection.Interior.Color = Clr
      InsertObj = Clr
End Function

```

Построить алгоритм и программу для параллельного заполнения нескольких упаковок.

### ***Вопросы к защите***

- 1 Какие переменные используются для управления процессом сортировки?
- 2 Каким образом построен алгоритм параллельного заполнения упаковок?

## **4 Построение модели и алгоритма управления работа-сборщика**

### **Задание**

По заданию, выданному преподавателем, построить алгоритм работы робота-сборщика, составить программу для его реализации.

По конвейеру в случайном порядке перемещаются заготовки  $N$  различных типов. Известна технологическая последовательность сборки.

Разработать алгоритм  $M$ -позиционной параллельной сборки изделия в технологической последовательности.

### ***Порядок выполнения работы***

Рабочее пространство робота содержит устанавливаемый параметр «Позиций для сборки» (рисунок 5). Информационные параметры – «Объем стека», «Собрано узлов», «Требуется», а также цветовая схема укомплектованных упаковок.

Схема взаимодействия элементов рабочего пространства представлена на рисунке 6.

На этой схеме элементы обозначены следующими именами (совпадающими с именами переменных в программе): Store – исходный накопитель или участок цеха, производящий детали по потребностям сборки; Conveyor – конвейер, подводящий детали к участку сборки; Positions – позиции сборки; Stacks –

промежуточные накопители-стеки для невостребованных в текущий момент деталей.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Конвейер	Деталь 1	Деталь 2	Деталь 3	Деталь 4	Деталь 5			Поз. 1	Поз. 2	Поз. 3	Поз. 4	Поз. 5	
2	113	5	6	8	3	0			1	5	5	5	5	Требуется
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														

ТАКТ →

СБРОС

Позиций для сборки: 5

Собрано узлов: 15

Рисунок 5 – Рабочее пространство работа

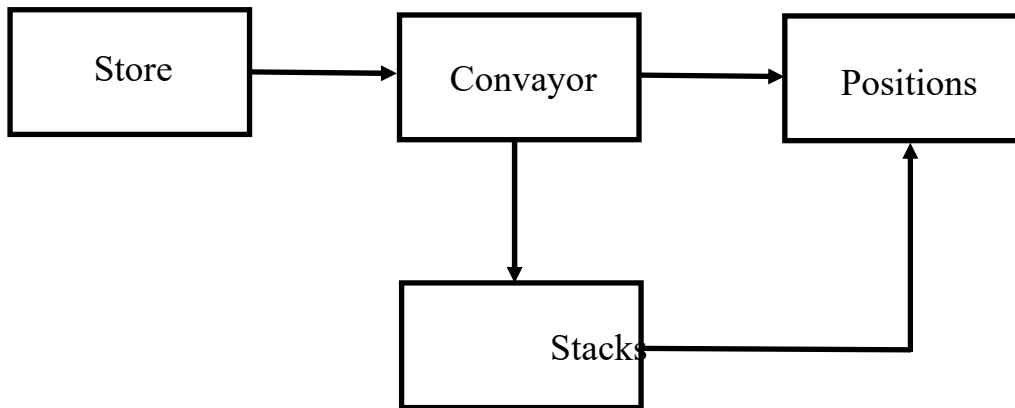


Рисунок 6 – Схема взаимодействия элементов рабочего пространства

Управляющий макрос для реализации такой схемы представлен ниже.

```
Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
```

```
Sub Main()
```

```
ans = Ins_stack_to_pos()
```

```
' перенести из стека (если он не пуст) на позицию
```

```
UnitIsReady ' отправить узел (если он собран)
```

```
Wait (500) ' ожидание
```

```
clr = Ins_store_to_konv()
```

```
' подать деталь из накопителя на конвейер
```

```
Wait (500)
```

```
ans = Ins_konv_to_pos(clr)
```

```
' подать деталь с конвейера (если требуется) на позицию
```

Wait (500)

If ans = 0 Then ans = Ins\_konv\_to\_stack()

' убрать деталь в накопитель (если использование на  
позиции ' невозможно)

End Sub

Sub UnitIsReady()

NPos = Range("h11") ' число позиций для сборки

For i = 1 To NPos

Need = Range("h2").Offset(0, i) ' какая деталь требуется

If Need = 6 Then ' убрать собранную позицию

Range("h2:h6").Offset(0, i).Select

Selection.Interior.Color = xlNone

Range("h2").Offset(0, i) = 1

NUnit = Range("h13") ' увеличить счетчик собранных узлов

Range("h13") = NUnit + 1

Exit For

End If

Next i

End Sub

Sub Wait(ms) ' ожидание

Debug.Print Now()

Sleep ms

End Sub

Function Ins\_store\_to\_konv()

lbl1: ans = Round(5 \* Rnd())

If ans = 0 Then GoTo lbl1

If ans = 1 Then clr = 255 ' Red

If ans = 2 Then clr = 65535 ' Yellow

If ans = 3 Then clr = 5287936 ' Green

If ans = 4 Then clr = 15773696 ' Blue

If ans = 5 Then clr = 10498160 ' Magenta

Range("a2").Select ' установить деталь на конвейер

Selection.Interior.Color = clr

Range("a2") = Range("a2") + 1

Ins\_store\_to\_konv = ans ' показать цвет



End Function

Function Ins\_konv\_to\_pos(ans)

```

If ans = 1 Then clr = 255      ' Red
If ans = 2 Then clr = 65535   ' Yellow
If ans = 3 Then clr = 5287936 ' Green
If ans = 4 Then clr = 15773696 ' Blue
If ans = 5 Then clr = 10498160 ' Magenta

```

```

NPos = Range("h11")

```

```

For i = 1 To NPos

```

```

    Need = Range("h2").Offset(0, i) ' какая деталь требуется

```

```

    If Need = ans Then

```

```

        Range("a2").Select ' убрать с конвейера
        Selection.Interior.Color = xlNone

```

```

        Range("h2").Offset(Need - 1, i).Select
        ' перенести на позицию сборки
        Selection.Interior.Color = clr

```

```

        Range("h2").Offset(0, i) = Need + 1
        ' требуется следующая деталь
        Ins_konv_to_pos = 1 ' сборка успешная
        Exit For

```

```

    Else

```

```

        Ins_konv_to_pos = 0 ' сборка не осуществлена

```

```

    End If

```

```

Next i

```

End Function

Function Ins\_konv\_to\_stack()

```

Range("a2").Select ' убрать деталь с конвейера
clr = Selection.Interior.Color
Range("a2").Interior.Color = xlNone

```

```

If clr = 255 Then Col = 1 ' определить номер стека

```

```

If clr = 65535 Then Col = 2

```

```

If clr = 5287936 Then Col = 3

```

```

If clr = 15773696 Then Col = 4

```

```

If clr = 10498160 Then Col = 5

```

```

Vol = Range("a2").Offset(0, Col)
' прочитать номер свободной позиции в стеке
Range("b2").Offset(Vol, Col - 1).Select ' перенести деталь в стек
Selection.Interior.Color = clr
Range("b2").Offset(0, Col - 1) = Vol + 1
' увеличить счетчик деталей в стеке

```

```

    Ins_konv_to_stack = 1
End Function

```

```

Function Ins_stack_to_pos()

```

```

    NPos = Range("h11")

```

```

    For i = 1 To NPos
        Need = Range("h2").Offset(0, i) ' требуется деталь
        If Need = 1 Then clr = 255      ' Red
        If Need = 2 Then clr = 65535   ' Yellow
        If Need = 3 Then clr = 5287936 ' Green
        If Need = 4 Then clr = 15773696 ' Blue
        If Need = 5 Then clr = 10498160 ' Magenta
    
```

```

        Vol = Range("a2").Offset(0, Need) ' объем стека
        If Vol <> 0 Then ' если стек не пуст
            Range("a2").Offset(0, Need) = Vol - 1
            ' уменьшить объем стека
            Range("a2").Offset(Vol - 1, Need).Select
            ' удалить деталь из стека
            Selection.Interior.Color = xlNone
        
```

```

            Range("h2").Offset(0, i) = Need + 1
            ' перенести деталь из стека на конвейер
            Range("h2").Offset(Need - 1, i).Select
            Selection.Interior.Color = clr
        
```

```

        End If
    Next i

```

```

End Function

```

```

Sub Reset() ' сброс и очистка рабочего поля
    Range("A2:F36").Select
    Selection.Interior.Pattern = xlNone

```

```
Range("I2:M6").Select
Selection.Interior.Pattern = xlNone
```

```
Range("B2").Select
ActiveCell.FormulaR1C1 = "0"
Selection.AutoFill Destination:=Range("B2:F2"),Type:=xlFillDefault
```

```
Range("I2").Select
ActiveCell.FormulaR1C1 = "1"
Selection.AutoFill Destination:=Range("I2:M2"),Type:=xlFillDefault
End Sub
```

По заданию преподавателя сформировать различные последовательности сборки, исследовать влияние параметров системы на ее производительность.

### ***Вопросы к защите***

- 1 Каким образом формируется последовательность параллельных заданий?
- 2 Какие показатели определяют производительность системы?

## **Список литературы**

- 1 **Масленникова, О. Е.** Основы искусственного интеллекта: учебное пособие / О. Е. Масленникова, И. В. Гаврилова. – 2-е изд., стер. – Москва: ФЛИНТА, 2013. – 282 с.
- 2 **Осипов, Г. В.** Методы искусственного интеллекта / Г. В. Осипов. – Москва: Физматлит, 2011. – 296 с.
- 3 **Рутковская, Д.** Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский. – Москва: Горячая линия-Телеком, 2007. – 452 с.
- 4 **Ярушкина, Н. Г.** Основы теории нечетких и гибридных систем: учебное пособие / Н. Г. Ярушкина. – Москва: Финансы и статистика, 2004. – 320 с.
- 5 **Нейронные сети. STATISTICA Neural Networks:** пер. с англ. – Москва: Горячая линия-Телеком, 2000. – 182 с.
- 6 **Биргер, И. А.** Техническая диагностика / И. А. Биргер. – Москва: Машиностроение, 1978. – 240 с.