```
1   PROGRAM Sim
2   VAR_INPUT
3       E3_CMD , //команда для включения распылителя 1 (зелёный)
4       E4_CMD , //команда для включения распылителя 2 (жёлтый)
5       MA2_UP_CMD , //команда для движения двери вверх
6       MA2_DOWN_CMD , //команда для движения двери вниз
7       E2_CMD , //команда для включения ультрафиолетовой лампы
8       MA1_LEFT_CMD , //команда для движения изделия влево
9       MA1_RIGHT_CMD , //команда для движения изделия вправо
10      HEAT_CMD , //команда для включения нагрева
11      MA4_UP_CMD , //команда для движения манипулятора вверх
12      MA4_DOWN_CMD , //команда для движения манипулятора вниз
13      MA3_LEFT_CMD , //команда для движения манипулятора влево
14      MA3_RIGHT_CMD , //команда для движения манипулятора вправо
15      MAN_LOCK_CMD , //команда на захват изделия
16      LAMP_Y , // индикация выбора программы 1 - покраска в зелёный
17      LAMP_G : BOOL ; // индикация выбора программы 1 - покраска в жёлтый
18      STEP : INT ; //номер шага автоматической программы
19      RUN : BOOL ; //индикатор запущенного технологического процесса - зелёный
    цвет в поле шага.
20  END_VAR
21  VAR
22      TEMP : REAL := 20 ; //значение температуры
23      ColorD : UDINT := 16#FFC0C0C0 ; //служебные переменные
24      ICollision , NVI : BOOL := 0 ; //служебные переменные
25      NI : BOOL := 1 ; //служебные переменные
26
27  END_VAR
28  VAR_OUTPUT
29          DoorFB , POS_IX , POS_MX , POS_MY , POS_IY : INT := 0 ; //служебные
    переменные
30          DoorErr , //служебные переменные //Сигнал авария двери
31          RESET_BTN , // кнопка сброса
32          START_BTN , // кнопка старта
33          PROG_SELECTOR : BOOL ; // переключатель для выбора программы если
    значение 0 - выбрана программа 1 (зелёный), если значение 1 - выбрана
    программа 2 (жёлтый)
34          TON2 : TON ; //служебные переменные
35          DOOR_POS_UP , //Сигнал дверь вверху
36          DOOR_POS_DOWN , //Сигнал дверь внизу
37          LOCK_ON_FB , //Сигнал изделие захвачено
38          IPOS_0 , IPOS_1 , IPOS_2 , IPOS_3 , IPOS_4 , IPOS_5 , // Сигналы
    изделие на позиции 0,1,2,3,4,5
39          I_err , //служебные переменные //Сигнал авария изделия
40          Man_err , //служебные переменные //Сигнал авария манипулятора
41          MPOS_0 , MPOS_1 , MPOS_2 , MPOS_UP , MPOS_DOWN : BOOL ; // Сигналы
    манимулятор на позиции 0,1,2,3,4,5
42          COUNT_Y , //служебные переменные //Количество выгруженных жёлтых
    изделий
43          COUNT_G : INT := 0 ; //служебные переменные //Количество выгруженных
    зелёных изделий
44  END_VAR
45
```

```
1   //22.09.2024
2   TON2 ( IN :=  ,  PT :=  ,  Q =>  ,  ET =>  ) ;
```

```
3       //движение двери вверз + анимация движения
4       IF  MA2_UP_CMD  THEN
5       DoorFB := DoorFB - 1 ;
6       END_IF
7
8       LAMP_Y := PROG_SELECTOR ;
9       LAMP_G := NOT  PROG_SELECTOR ;
10
11      //движение двери вниз + анимация движения
12      IF  MA2_DOWN_CMD  THEN
13      DoorFB := DoorFB + 1 ;
14      END_IF
15
16      // Ошибка двери
17      IF  DoorFB  <= - 123
18      OR  DoorFB  > 0  THEN
19      DoorErr := 1 ;
20      END_IF
21
22      // Имитация датчиков конечного положения для двери
23      IF  DoorFB  < - 110  THEN
24          DOOR_POS_UP := 1 ;
25          ELSE
26          DOOR_POS_UP := 0 ;
27      END_IF
28
29      IF  DoorFB  > - 4  THEN
30          DOOR_POS_DOWN := 1 ;
31          ELSE
32          DOOR_POS_DOWN := 0 ;
33      END_IF
34
35      //Имитация нагрева
36      IF  HEAT_CMD  THEN
37      TEMP := TEMP + 0.1 ;
38      IF  TEMP  >  160  THEN
39          TEMP := 160 ;
40          END_IF ;
41      ELSIF   TEMP > 20  THEN
42          TEMP := TEMP - 0.05 ;
43      END_IF
44
45      // Реакция на сброс
46      IF  RESET_BTN  THEN
47          STEP := 0 ;
48          COUNT_Y := COUNT_G := 0 ;
49          NVI := 0 ;
50          TEMP := 20 ;
51          ColorD := 16#FFC0C0C0 ;
52          TON2 . IN := 0 ;
53          POS_IX := 0 ;
54          POS_IY := 0 ;
55          POS_MY := 0 ;
56          POS_MX := 0 ;
57          DoorFB := 0 ;
58          DoorErr := 0 ;
```

```
59          I_err := 0 ;
60          Man_err := 0 ;
61          RUN := 0 ;
62      END_IF
63
64      //Отслеживаем покраску 1
65      IF       E3_CMD = 1  AND  IPOS_2  THEN
66               TON2 . PT := T#3S ;
67               TON2 . IN := 1 ;
68               IF  TON2 . Q  THEN
69                   ColorD := 16#FF008000 ;
70                   E3_CMD := 0 ;
71               END_IF
72      END_IF
73
74      //Отслеживаем покраску 2
75      IF       E4_CMD = 1  AND  IPOS_3  THEN
76               TON2 . PT := T#3S ;
77               TON2 . IN := 1 ;
78               IF  TON2 . Q  THEN
79                   ColorD := 16#FFFFFF00 ;
80                   E4_CMD := 0 ;
81               END_IF
82      END_IF
83
84      IF  NOT  E3_CMD  AND  NOT  E4_CMD  THEN
85                       TON2 . IN := 0 ;
86      END_IF
87
88      //Отслеживаем положение двери и изделия
89      IF  ( POS_IX  >=  5  AND  POS_IX  <=  88  AND  DoorFB  > - 82 )
90        THEN
91      ICollision := 1 ;
92      ELSE
93      ICollision := 0 ;
94      END_IF
95
96      //Двигаем изделие
97      IF  MA1_RIGHT_CMD  AND  ( NOT  ICollision  OR  ( ICollision  AND  POS_IX  >  87 ) )
          THEN
98      POS_IX := POS_IX + 1 ;
99      END_IF
100
101     IF  MA1_LEFT_CMD  AND  ( NOT  ICollision  OR  ( ICollision  AND  POS_IX  <  6  ) )
        THEN
102     POS_IX := POS_IX - 1 ;
103     END_IF
104
105     //Имитируем датчики конечного положения для изделия
106     IF  POS_IX  > - 4  AND  POS_IX  < 4  THEN
107         IPOS_0 := 1 ;
108         ELSE
109         IPOS_0 := 0 ;
110     END_IF
111
112     IF  POS_IX  > 170  AND  POS_IX  < 178  THEN
```

```
113             IPOS_1 := 1 ;
114             ELSE
115             IPOS_1 := 0 ;
116         END_IF
117
118         IF  POS_IX  > 340  AND  POS_IX  < 348  THEN
119             IPOS_2 := 1 ;
120             ELSE
121             IPOS_2 := 0 ;
122         END_IF
123
124         IF  POS_IX  > 418  AND  POS_IX  < 426  THEN
125             IPOS_3 := 1 ;
126             ELSE
127             IPOS_3 := 0 ;
128         END_IF
129
130         IF  POS_IX  > 524  AND  POS_IX  < 532  THEN
131             IPOS_4 := 1 ;
132             ELSE
133             IPOS_4 := 0 ;
134         END_IF
135
136         IF  POS_IX  > 636  AND  POS_IX  < 644  AND  POS_IY  >  - 4  THEN
137             IPOS_5 := 1 ;
138             ELSE
139             IPOS_5 := 0 ;
140         END_IF
141
142
143
144         //Пробуем захватить изделие
145         IF  IPOS_5  AND  MAN_LOCK_CMD  THEN
146             LOCK_ON_FB := 1 ;
147         END_IF
148
149         IF  NOT  MAN_LOCK_CMD  THEN
150             LOCK_ON_FB := 0 ;
151         END_IF
152
153         //Управляем манипулятором
154         IF  LOCK_ON_FB  THEN
155             IF  MA4_UP_CMD  THEN
156                 POS_IY := POS_IY - 1 ;
157                 POS_MY := POS_MY - 1 ;
158             END_IF
159             IF  MA4_DOWN_CMD  THEN
160                 POS_IY := POS_IY + 1 ;
161                 POS_MY := POS_MY + 1 ;
162             END_IF
163             IF  MA3_RIGHT_CMD  THEN
164                 POS_MX := POS_MX + 1 ;
165                 POS_IX := POS_IX + 1 ;
166             END_IF
167             IF  MA3_LEFT_CMD  THEN
168                 POS_MX := POS_MX - 1 ;
```

```
169             POS_IX := POS_IX - 1 ;
170         END_IF
171         ELSE
172         IF  MA4_UP_CMD  THEN
173             POS_MY := POS_MY - 1 ;
174         END_IF
175         IF  MA4_DOWN_CMD  THEN
176             POS_MY := POS_MY + 1 ;
177         END_IF
178         IF  MA3_RIGHT_CMD  THEN
179             POS_MX := POS_MX + 1 ;
180         END_IF
181         IF  MA3_LEFT_CMD  THEN
182             POS_MX := POS_MX - 1 ;
183         END_IF
184     END_IF
185
186     //Ошибка при перемещении манипулятора за предельные границы
187     IF  POS_MX  > 351  OR  POS_MX  < - 15  THEN
188         Man_err := 1 ;
189     END_IF
190
191     IF  POS_MY  > 11  OR  POS_MY  < - 193  THEN
192         Man_err := 1 ;
193     END_IF
194
195     //Имитируем датчики конечного положения для манипулятора
196     IF  POS_MX  > - 2  AND  POS_MX  < 2  THEN
197         MPOS_0 := 1 ;
198         ELSE
199         MPOS_0 := 0 ;
200     END_IF
201
202     IF  POS_MX  > 210  AND  POS_MX  < 214  THEN
203         MPOS_1 := 1 ;
204         ELSE
205         MPOS_1 := 0 ;
206     END_IF
207
208     IF  POS_MX  > 320  AND  POS_MX  < 326  THEN
209         MPOS_2 := 1 ;
210         ELSE
211         MPOS_2 := 0 ;
212     END_IF
213
214     IF  POS_MY  > - 134  AND  POS_MY  < - 128  THEN
215         MPOS_UP := 1 ;
216         ELSE
217         MPOS_UP := 0 ;
218     END_IF
219
220     IF  POS_MY  > - 7  AND  POS_MY  < 1  THEN
221         MPOS_DOWN := 1 ;
222         ELSE
223         MPOS_DOWN := 0 ;
224     END_IF
```

```
225
226     //Проверяем и фиксируем выгрузку
227     IF NOT LOCK_ON_FB THEN
228             IF POS_IX > 840 AND POS_IX < 850 AND POS_IY < 0 THEN
229             COUNT_G := COUNT_G + 1 ;
230             POS_IX := - 10 ;
231             NVI := 1 ;
232             END_IF
233             IF POS_IX > 950 AND POS_IX < 960 AND POS_IY < 0 THEN
234             COUNT_Y := COUNT_Y + 1 ;
235             POS_IX := - 10 ;
236             NVI := 1 ;
237             END_IF
238     END_IF
239
240     //Ошибка при перемещении изделия за предельные границы
241     IF ( POS_IX > 679 AND NOT LOCK_ON_FB ) OR POS_IX < - 21 THEN
242         I_err := 1 ;
243     END_IF
244     //Для проверки
245     // Sim(
246     //         Color1CMD:=Sim.Color1BTN ,
247     //         Color2CMD:= Sim.Color2BTN,
248     //         DoorUpCMD:= Sim.DoorUpBTN,
249     //         DoorDownCMD:= Sim.DoorDownBTN,
250     //         UFLampCMD:= Sim.UFLampBTN,
251     //         MotorLeftCMD:= Sim.MotorLeftBTN,
252     //         MotorRightCMD:= Sim.MotorRigthBTN,
253     //         RunCMD:= Sim.StartBTN,
254     //         HeatCMD:= Sim.HeatBTN,
255     //         ManUpCMD:= Sim.ManUpBTN,
256     //         ManDownCMD:= Sim.ManDownBTN,
257     //         ManLeftCMD:= Sim.ManLeftBTN,
258     //         ManRightCMD:= Sim.ManRigthBTN,
259     //         LampY:= Sim.ProgSelector,
260     //         LampG:= NOT Sim.ProgSelector,
261     // );
262
```