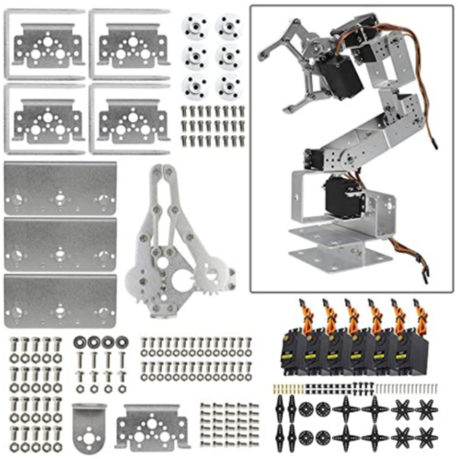
# Chapter 31 Simple Robot

## Lab 13.4 Simple Robotic Arm

Using the program from Ch. 8 for the servo arm, design a robotic arm that has four degrees of motion. The arm is to move in a set number of motions with a time delay between motions so as to pick up a coin and lay it down in another spot.

The four axes control the four servo motors in the robot with PWM output control. The processor is the S7-1215 DCDCDC model. The axes are set up similar to the code below. The program can be written to move between various settings for the sp_x variables for the four axes.


Due to the weight of the end effector, the 5th and 6th servos are removed. The end of arm tool is replaced with the suction device shown below. This is attached to servo 4 shown at right.



# diymore Silver ROT3U 6DOF Aluminium Robot Arm Mechanical Robotic Clamp Claw Kit with MG996R Servos 25T Metal Disc Horns and Screw (Unassembled)

Visit the diymore Store

5 ratings

$82.99

prime

& FREE Returns

Save up to 6% with business pricing. Sign up for free Amazon Business account

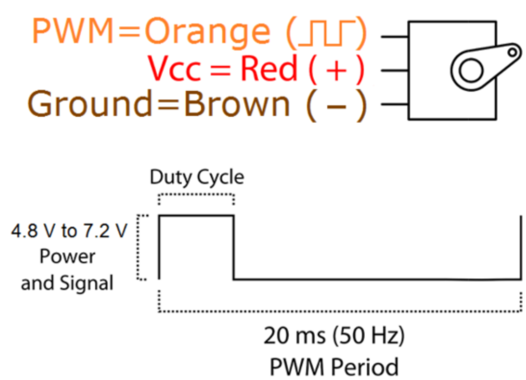
Color: **Silver**

$82.88 prime | $82.99 prime

Roll over image to zoom in

- Radius of gyration: 355mm.
- Rotation angle of 180 degrees.
- Holder of the widest distance: 98mm.
- If the item doesn't come with the guide/manual, so please kindly contact us for help. We will send you detailed user manual.
- You need to assemble clamp claw kits by yourself. You'd better use MG996R servos for the joint bears larger force, we provide everything you need during the installation. (MG996R servos and 25T metal disc horns)

Specifications for the MG996R Servo follow. The high current requires a significant 5V power supply. The Lambda 5V-5A supply shown lower right is sufficient for the load.

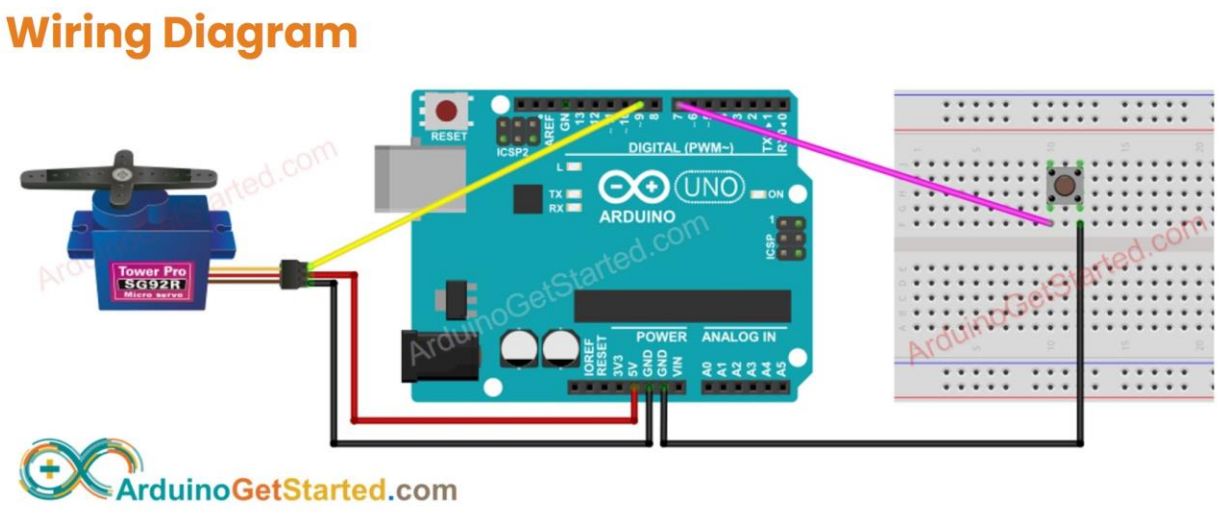



Specifications of Servo

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 9.4 kgf·cm (4.8 V ), 11 kgf·cm (6 V)
- Operating speed: 0.17 s/60º (4.8 V), 0.14 s/60º (6 V)
- Operating voltage: 4.8 V a 7.2 V
- Running Current 500 mA – 900 mA (6V)
- Stall Current 2.5 A (6V)
- Dead band width: 5 µs
- Stable and shock proof double ball bearing design
- Temperature range: 0 ºC – 55 ºC

Lambda 5V Power Supply

We have four PWM outputs from the PLC but we need a fifth servo controller to control the lynxmotion vacuum end of arm tool.  The best way to control it is to use an Arduino with a digital output connected from the PLC to the Arduino and the Arduino directly connected to the servo controlling the vacuum tube.  The picture for the connection is similar to the button connection shown below:

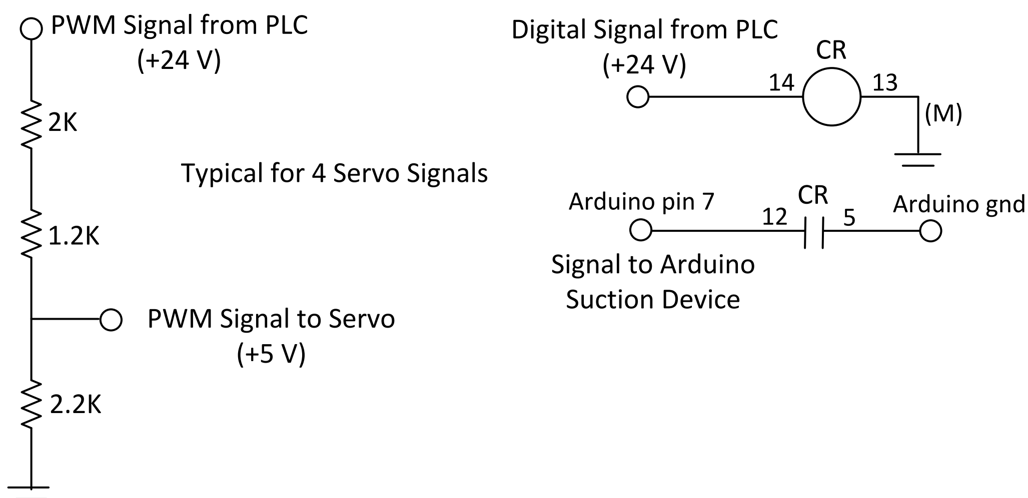



Code for the Arduino is as follows:

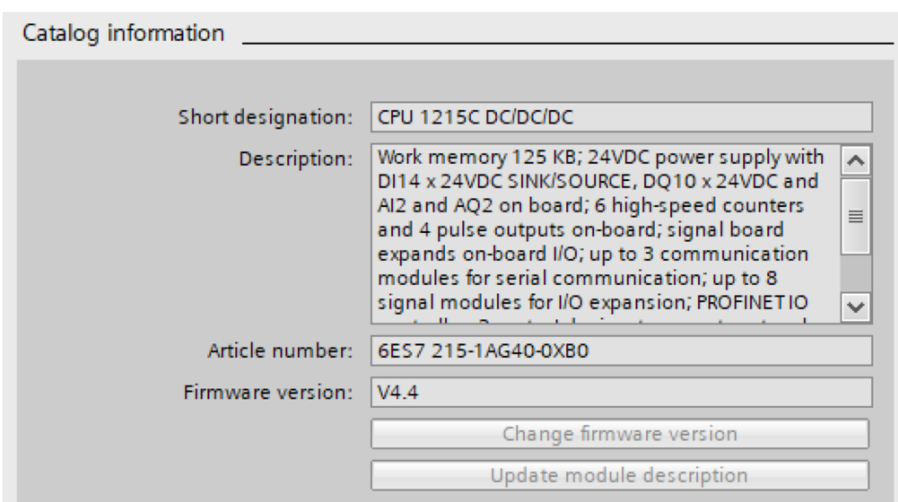
```
#include <Servo.h>

// constants won't change
const int BUTTON_PIN = 7; // Arduino pin connected to button's pin
const int SERVO_PIN  = 9; // Arduino pin connected to servo motor's pin

Servo servo; // create servo object to control a servo

// variables will change:
int angle = 0;          // the current angle of servo motor
int currentButtonState; // the current state of button

void setup() {
  pinMode(BUTTON_PIN, INPUT_PULLUP); // set arduino pin to input pull-up mode
  servo.attach(SERVO_PIN);           // attaches the servo on pin 9 to the
   servo object

  servo.write(angle);
 }

void loop() {

  currentButtonState = digitalRead(BUTTON_PIN); // read new state

  if(currentButtonState == LOW)
```

```
    // change angle of servo motor
    angle = 90;
    else
    angle = 0;

    // control servo motor arccoding to the angle
    servo.write(angle);
 }
```

Wiring for the four servo signals and the suction device is as follows:




Configuration of the S7-1215 follows:


Catalog information

Short designation: CPU 1215C DC/DC/DC

Description: Work memory 125 KB; 24VDC power supply with DI14 x 24VDC SINK/SOURCE, DQ10 x 24VDC and AI2 and AQ2 on board; 6 high-speed counters and 4 pulse outputs on-board; signal board expands on-board I/O; up to 3 communication modules for serial communication; up to 8 signal modules for I/O expansion; PROFINET IO

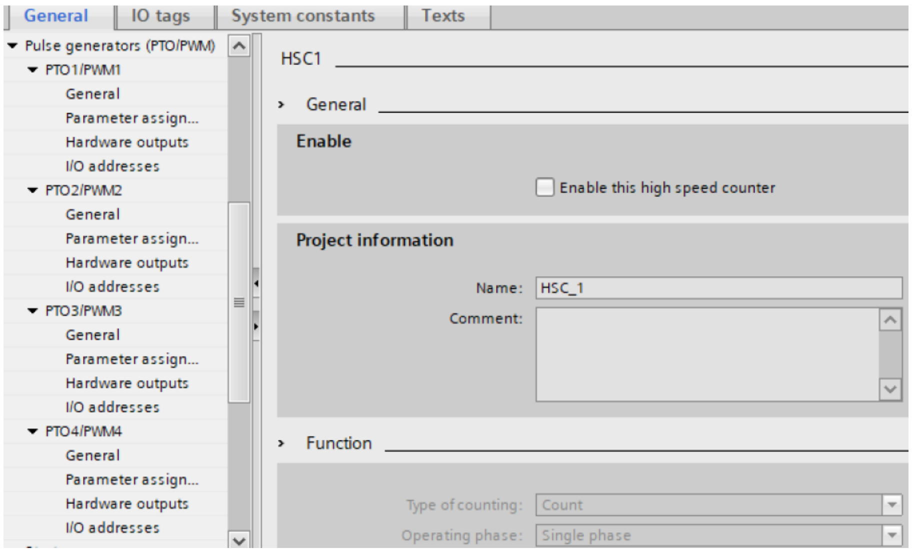
Article number: 6ES7 215-1AG40-0XB0

Firmware version: V4.4

Change firmware version

Update module description

The PWM outputs are controlled through configuring PWM1, PWM2, PWM3 and PWM4 below:

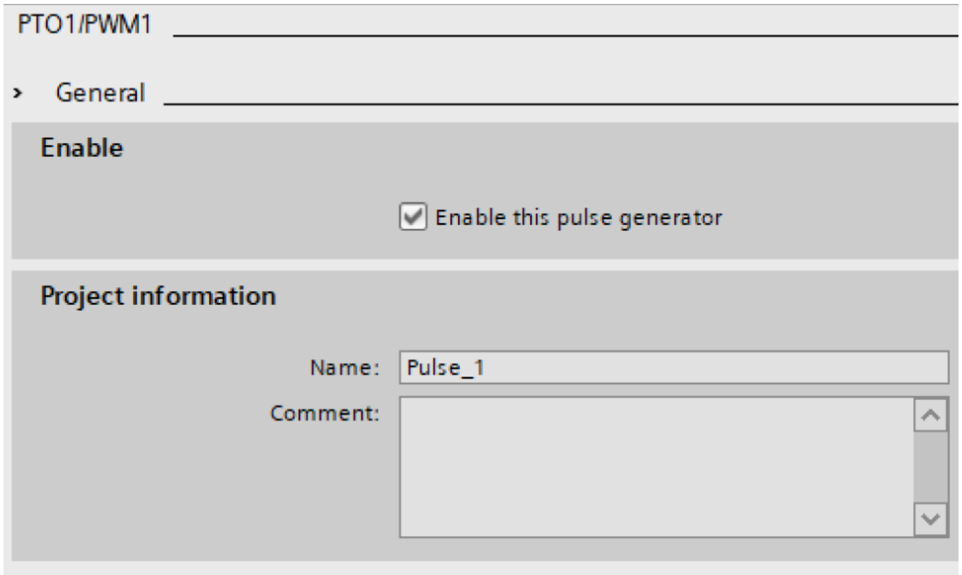


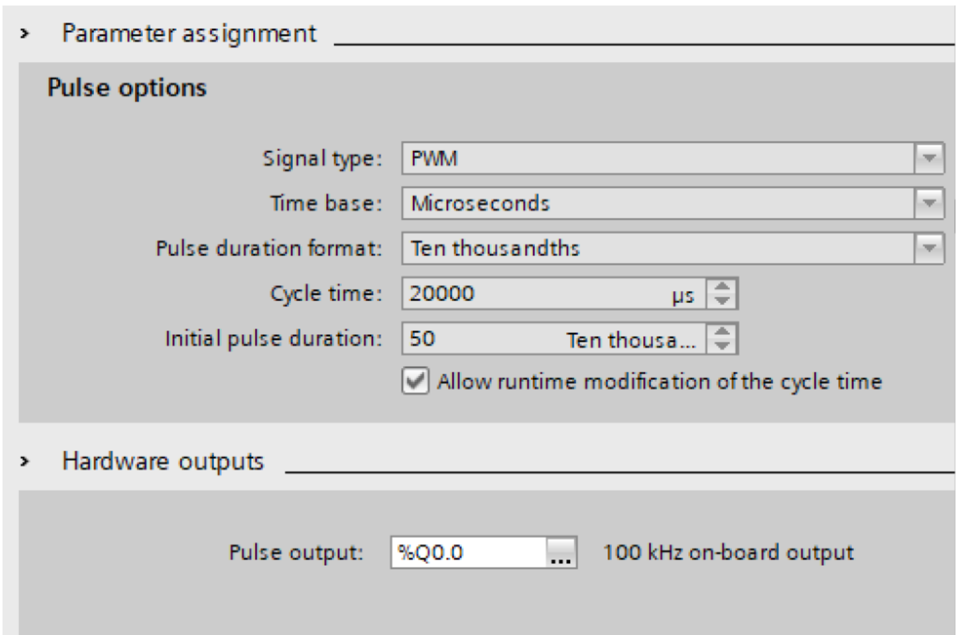

Variables that control movement are sp_1, 2, 3, and 4 (setpoints). The program may be tested by setting ena to 1 and test_1, 2, 3, and 4 to 1. The setpoint variables are the position points.

Project3 ▸ PLC_1 [CPU 1215C DC/DC/DC] ▸ Watch and force tables ▸ Watch table_1

| | i | Name | Address | Display format | Monitor value | Modify value | | | Comment |
|---|---|---|---|---|---|---|---|---|---|
| 1 | | "Tag_1" | %QW1008 | DEC+/- | | 700 | ☑ | ⚠ | |
| 2 | | "rate_1" | %MD2 | Time | | T#25MS | ☑ | ⚠ | |
| 3 | | "sp_1" | %MW10 | DEC+/- | | 700 | ☑ | ⚠ | |
| 4 | | "sp_2" | %MW12 | DEC+/- | | 500 | ☑ | ⚠ | |
| 5 | | "sp_3" | %MW14 | DEC+/- | | 700 | ☑ | ⚠ | |
| 6 | | "sp_4" | %MW20 | DEC+/- | | 400 | ☑ | ⚠ | |
| 7 | | | <Add new> | | | | ☐ | | |

Configuration of the PWM axes follow. They all resemble PWM1 shown below:

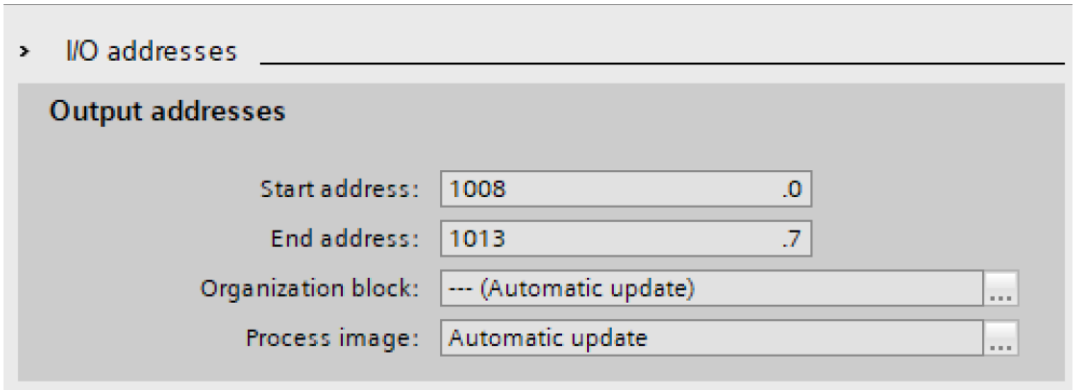



The other outputs are wired as follows:

Axis 2 – Q0.1
Axis 3 – Q0.3
Axis 4 – Q0.4

(These values could have been Q0.0, Q0.1, Q0.2 and Q0.3 but it was found that the second output had some problems with a RC time constant and the waveform was not squared as with the other points.  Use of an oscilloscope is extremely valuable in troubleshooting the waves.)



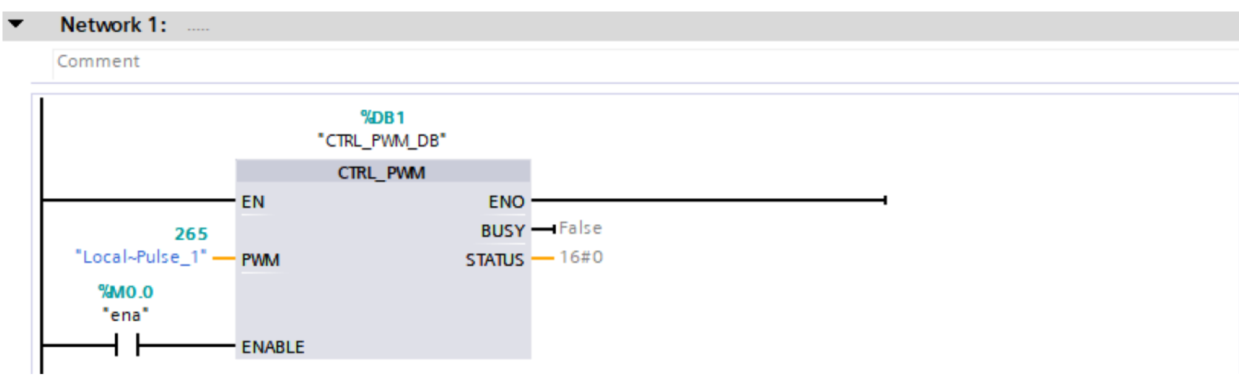

Output variables to be written to are:

Axis 1 - QW1008
Axis 2 – QW2
Axis 3 - QW1014
Axis 4 - QW8

Each variable is 16 bit (Word length).

Each axis must contain the following logic:

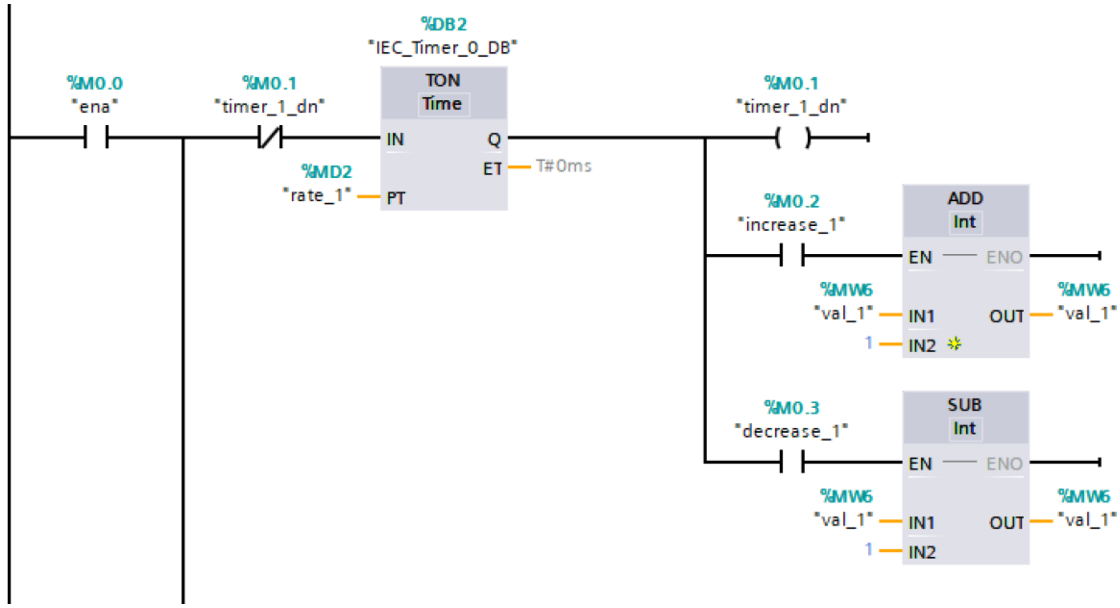
This instruction must be inserted to set up the PWM motion.

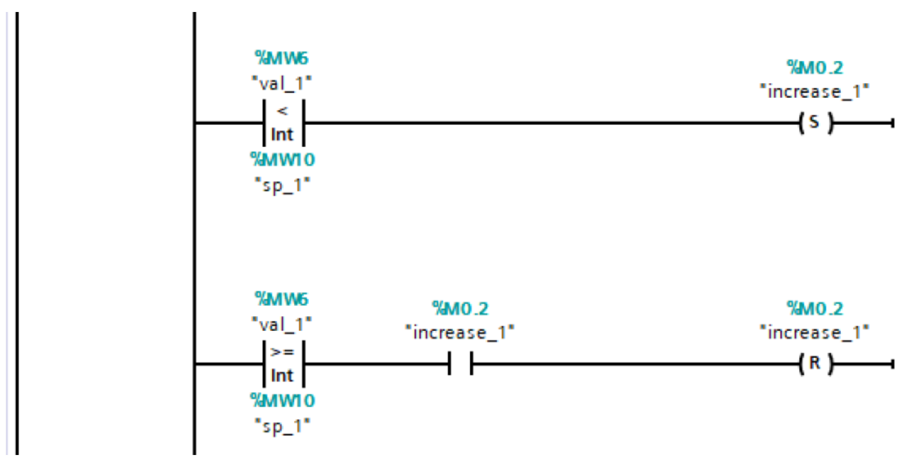


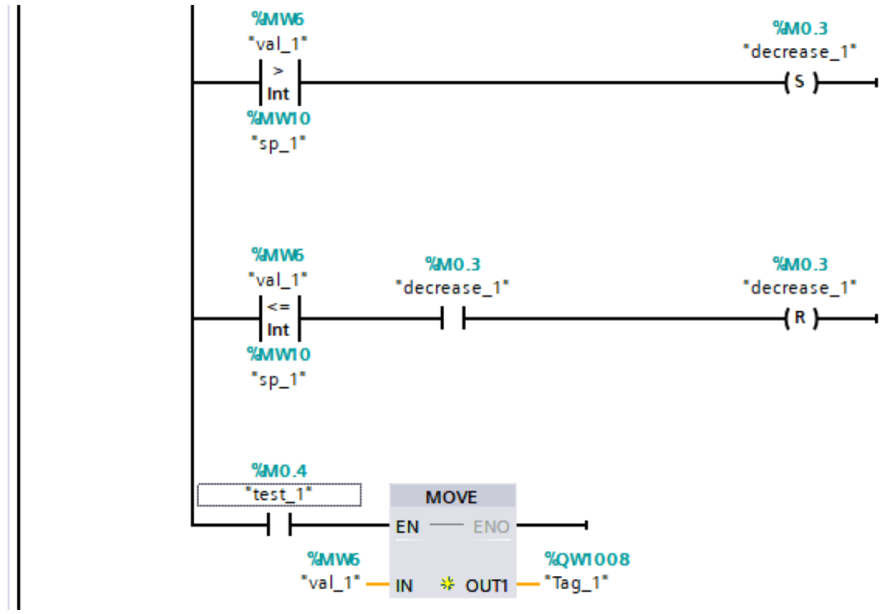

This lab may be programmed several different ways. The best would include arrays that have positional data for the various axes. As the robot moves through various positions, the robot program moves down through the list of position data at each position. There should also be a rate of movement for each of the axes (or perhaps just one common rate as programmed here). The rate of movement is important as well as sequence of the positional data.

**At present, only one of these robotic arms is available. The 5 V separate power supply is extremely important as the current draw by the various axes is large.**

The following logic is a timer that either increases or decreases the value to be output to the PWM. The setpoint determines whether the value is to grow or decrease. This code is repeated for each of the four axes. To save movements, begin to think about construction of an array or UDT. Either will allow a sequence of steps that can be programmed to control the movement.



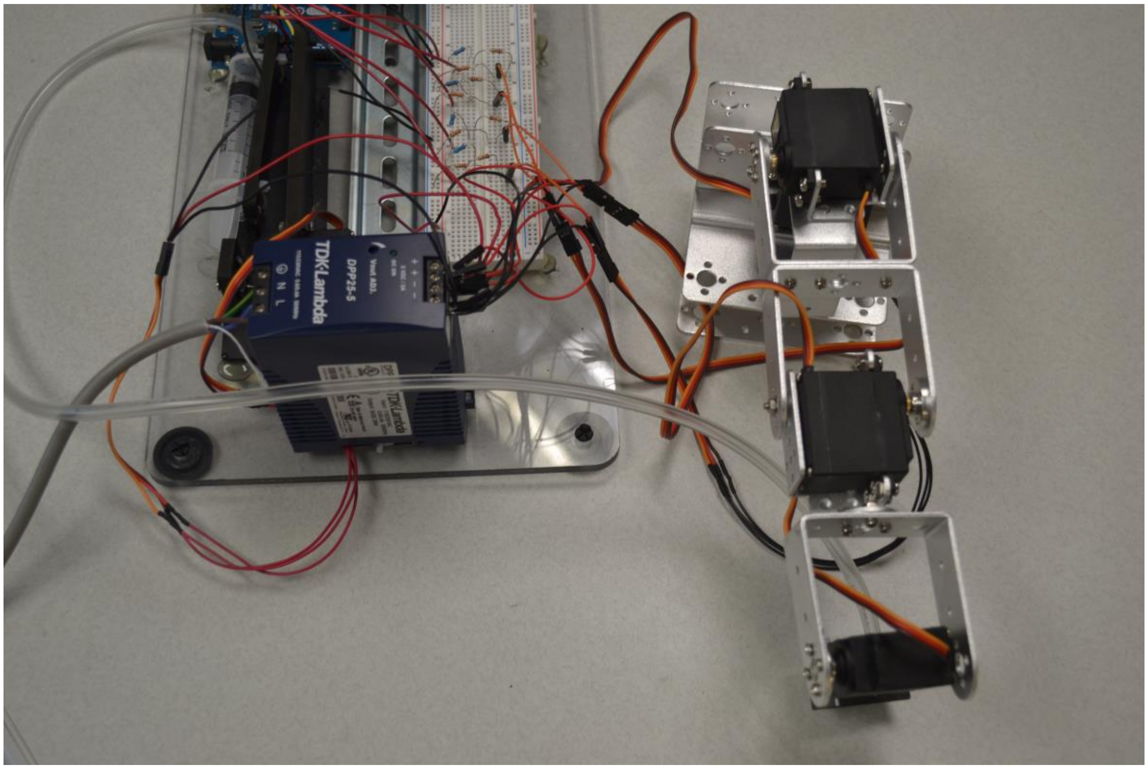



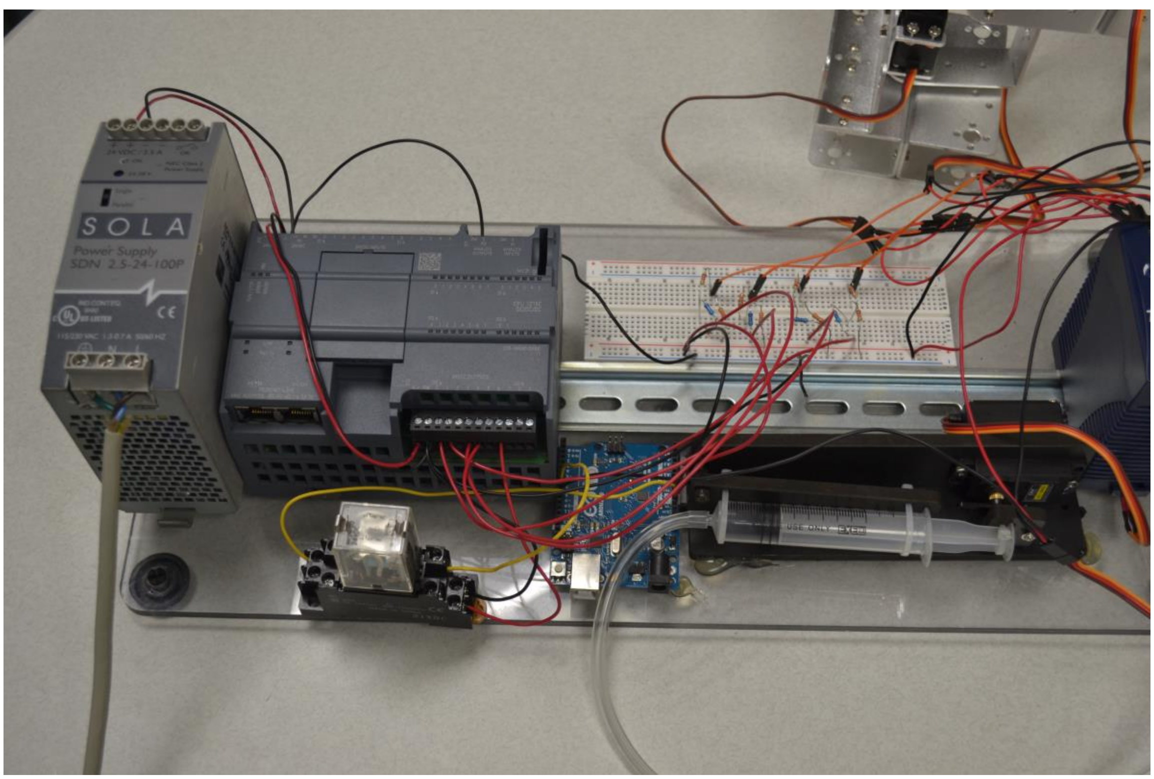


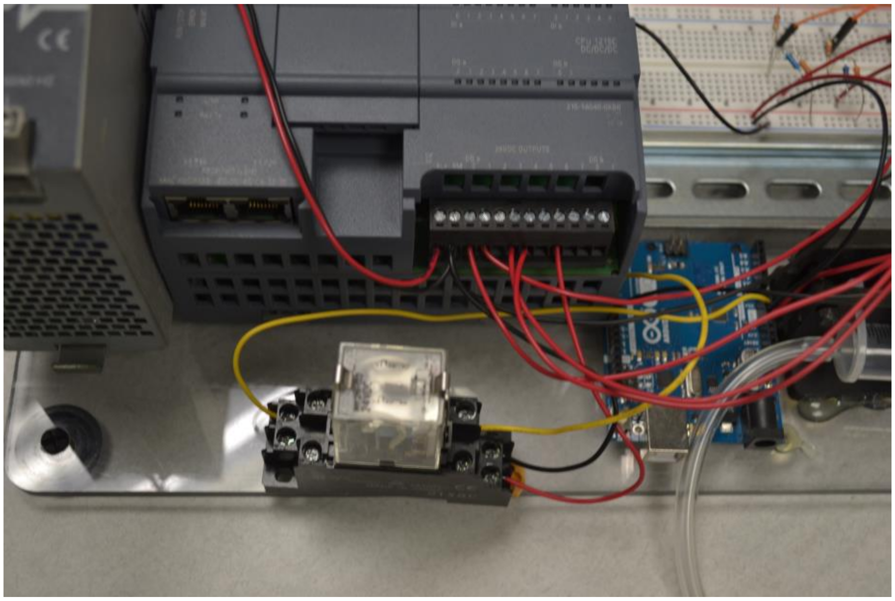

For testing purposes, construct a watch table to change various axes' setpoints.

| | i | Name | Address | Display format | Monitor value | Modify value | | |
|---|---|---|---|---|---|---|---|---|
| 1 | | "Tag_1" | %QW1008 | DEC+/- | | 400 | ☑ | ⚠ |
| 2 | | "rate_1" | %MD2 | Time | | T#25MS | ☑ | ⚠ |
| 3 | | "sp_1" | %MW10 | DEC+/- | | 900 | ☑ | ⚠ |
| 4 | | "sp_2" | %MW12 | DEC+/- | | 1100 | ☑ | ⚠ |
| 5 | | "sp_3" | %MW14 | DEC+/- | | 600 | ☑ | ⚠ |
| 6 | | "sp_4" | %MW20 | DEC+/- | | 700 | ☑ | ⚠ |
| 7 | | | %Q0.6 | Bool | | TRUE | ☑ | ⚠ |

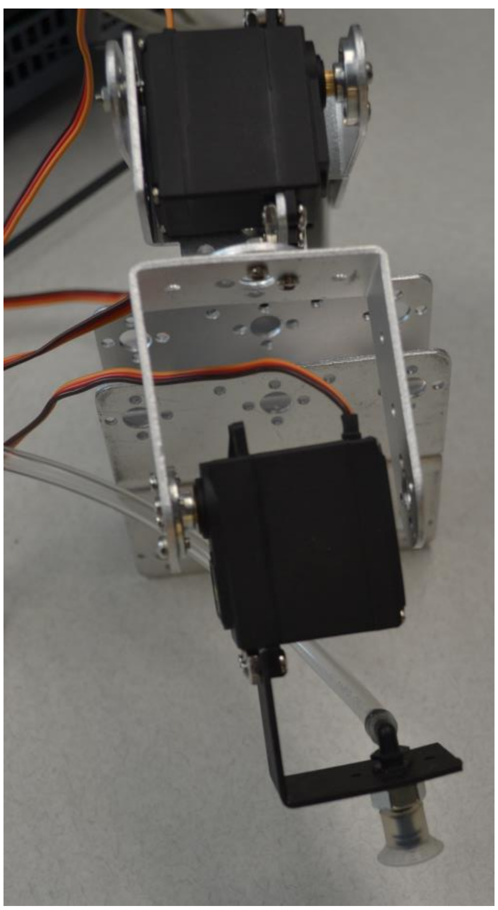
The following pictures show the construction of the robot with a breadboard for resistor values.

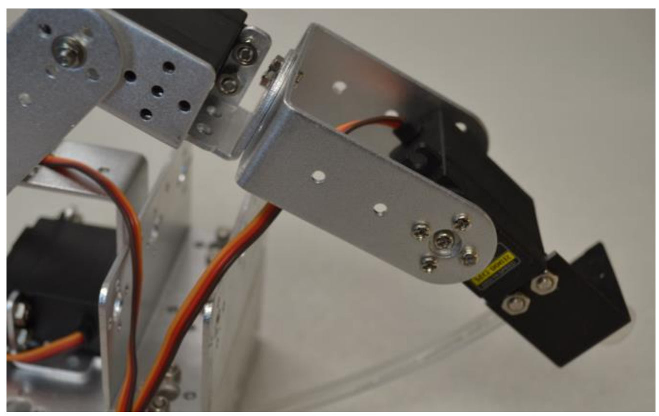




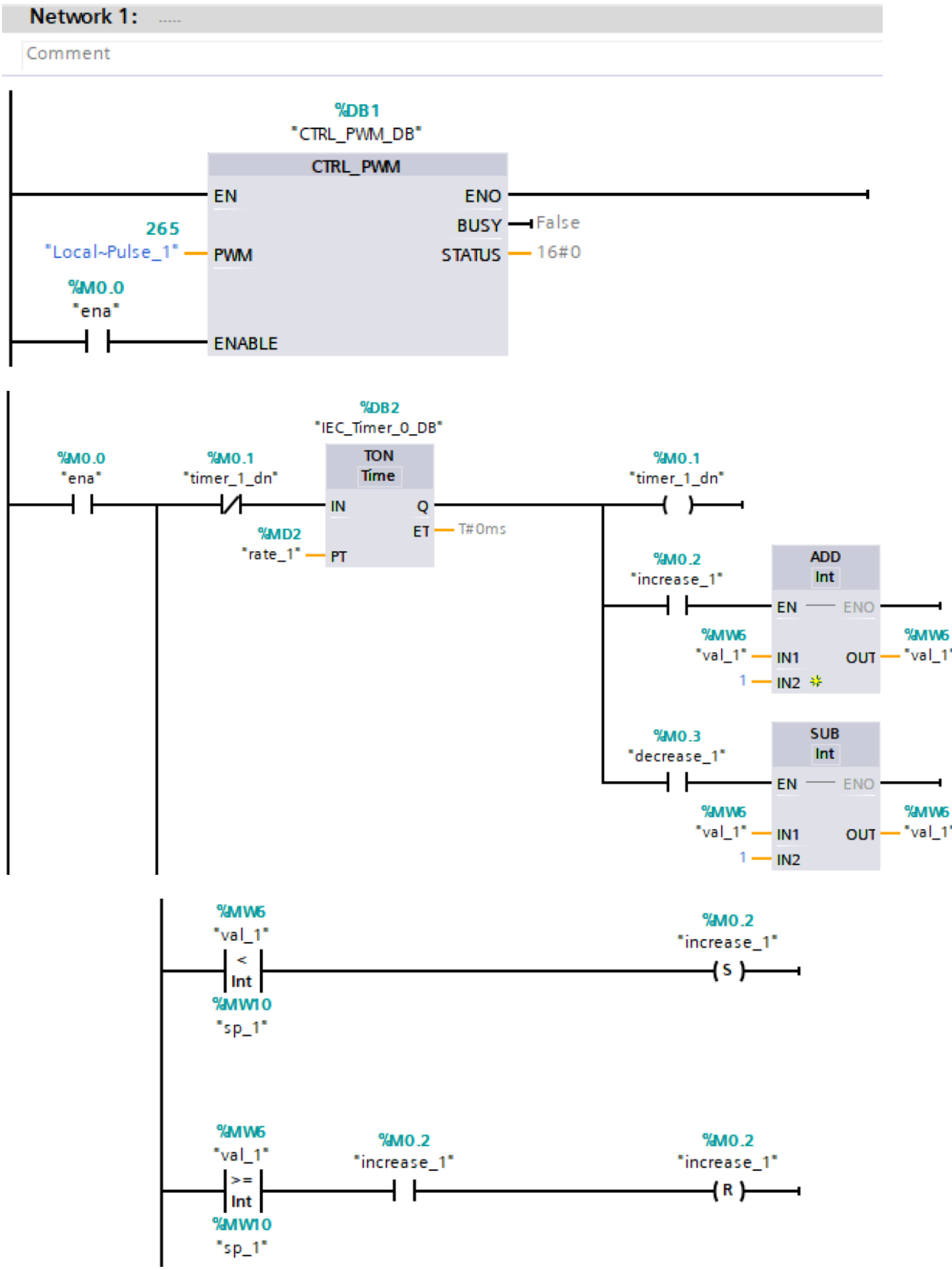
This lab works!  It allows the construction of a moving robot from the PLC.  It also encourages the construction and use of arrays for recipe storage.  Additional labs may be made to set up the movements and program the robot as well as just control the movements via set program statements.

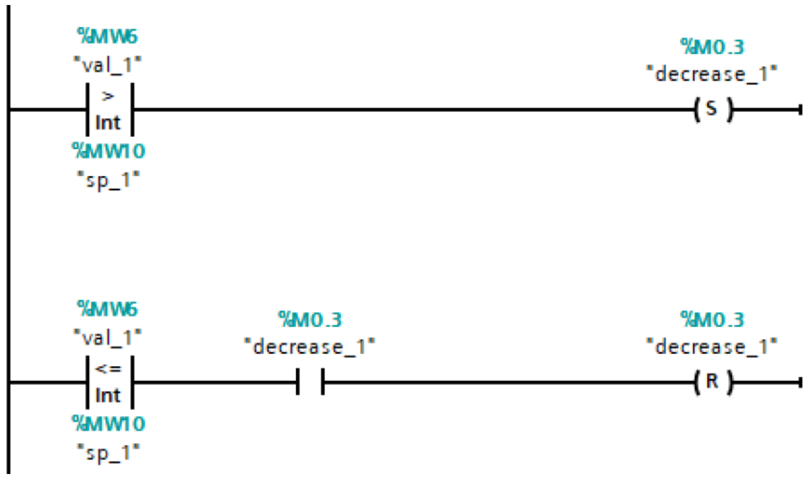


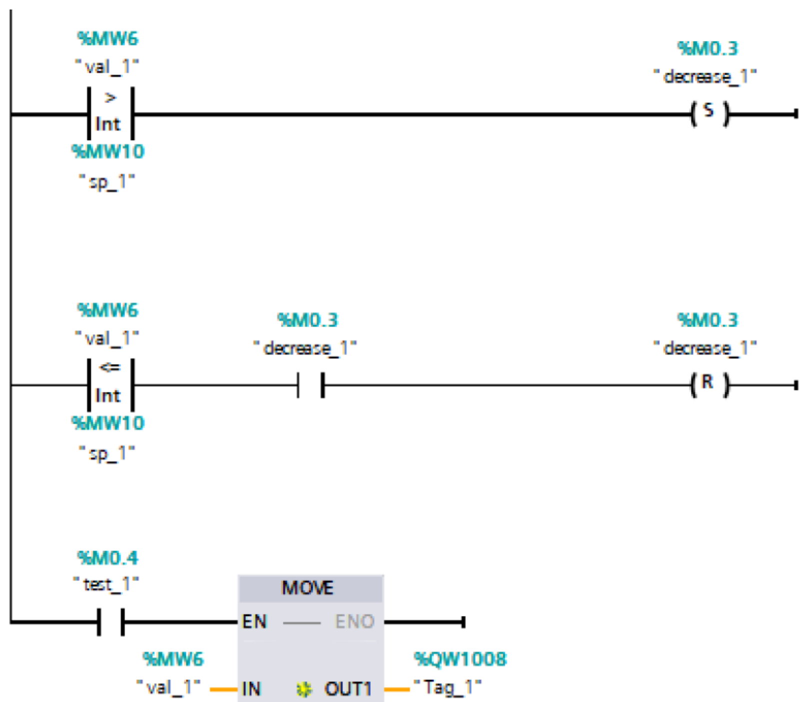


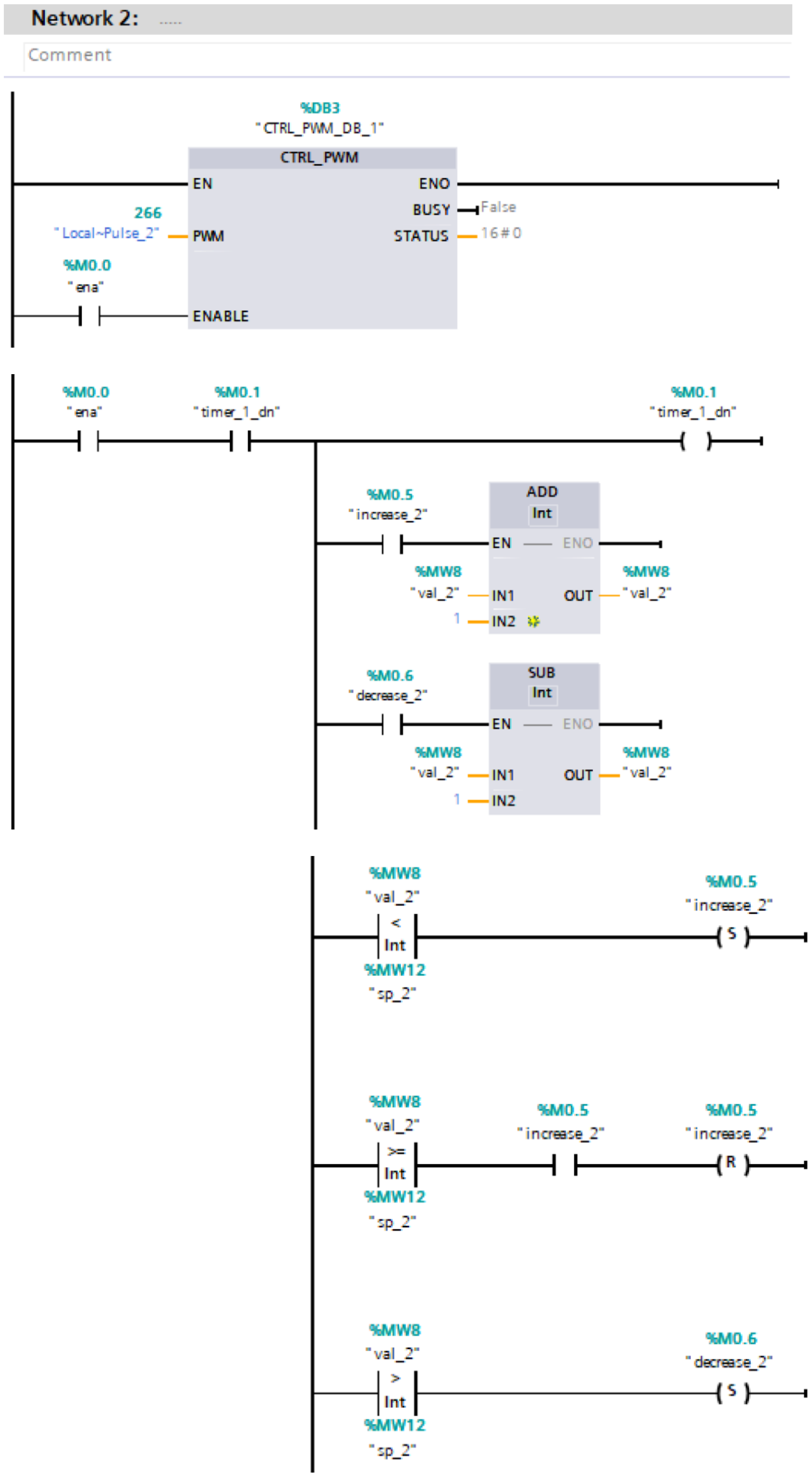
Note that servo only attached on left side

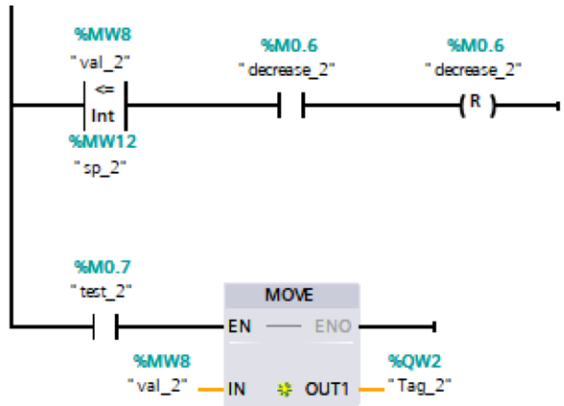
End Tool – Modified from original


Network 1:
.....
Comment
%DB1
"CTRL_PWM_DB"
CTRL_PWM
EN
ENO
BUSY
False
265
"Local~Pulse_1"
PWM
STATUS
16#0
%M0.0
"ena"
ENABLE
%DB2
"IEC_Timer_0_DB"
%M0.0
"ena"
%M0.1
"timer_1_dn"
TON
Time
IN
Q
ET
T#0ms
%MD2
"rate_1"
PT
%M0.1
"timer_1_dn"
%M0.2
"increase_1"
ADD
Int
EN
ENO
%MW6
"val_1"
IN1
OUT
%MW6
"val_1"
1
IN2
%M0.3
"decrease_1"
SUB
Int
EN
ENO
%MW6
"val_1"
IN1
OUT
%MW6
"val_1"
1
IN2
%MW6
"val_1"
<
Int
%MW10
"sp_1"
%M0.2
"increase_1"
S
%MW6
"val_1"
>=
Int
%MW10
"sp_1"
%M0.2
"increase_1"
%M0.2
"increase_1"
R


%MW6
"val_1"
>
Int
%MW10
"sp_1"
%M0.3
"decrease_1"
S
%MW6
"val_1"
<=
Int
%MW10
"sp_1"
%M0.3
"decrease_1"
%M0.3
"decrease_1"
R

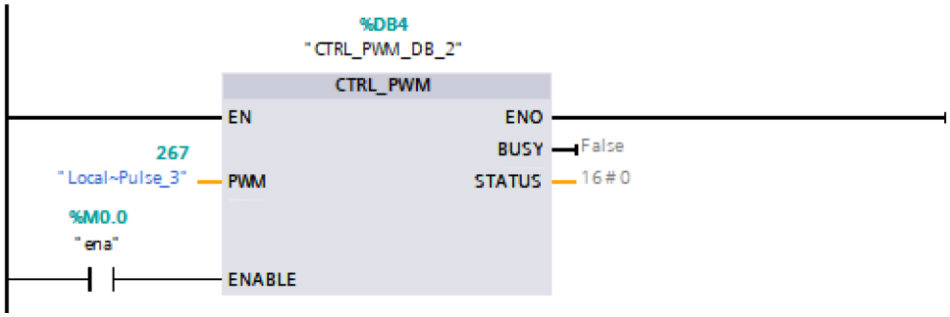

%MW6
"val_1"
>
Int
%MW10
"sp_1"
%M0.3
"decrease_1"
S
%MW6
"val_1"
<=
Int
%MW10
"sp_1"
%M0.3
"decrease_1"
%M0.3
"decrease_1"
R
%M0.4
"test_1"
MOVE
EN
ENO
%MW6
"val_1"
IN
OUT1
%QW1008
"Tag_1"

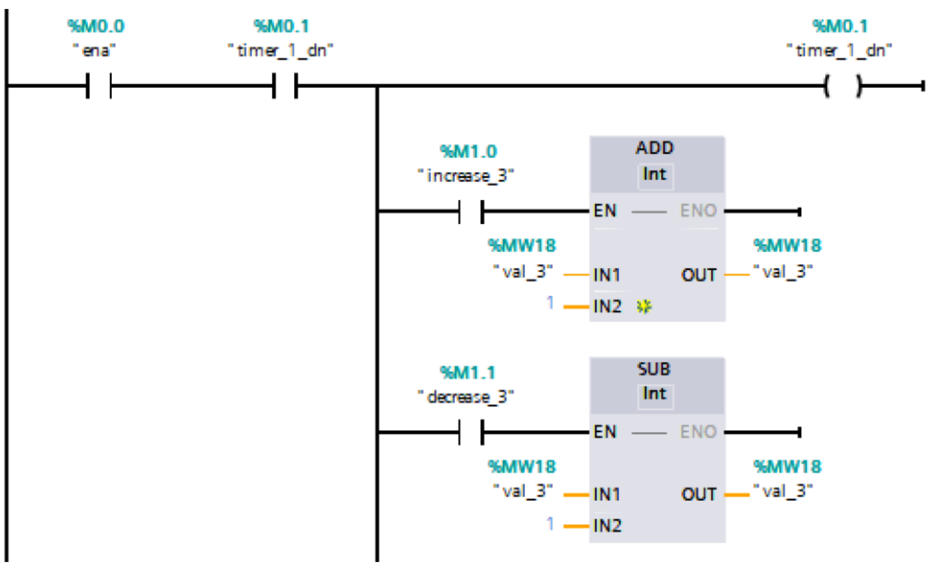

Network 2:
Comment
%DB3
"CTRL_PWM_DB_1"
CTRL_PWM
EN
ENO
BUSY
False
266
"Local~Pulse_2"
PWM
STATUS
16#0
%M0.0
"ena"
ENABLE
%M0.0
"ena"
%M0.1
"timer_1_dn"
%M0.1
"timer_1_dn"
%M0.5
"increase_2"
ADD
Int
EN
ENO
%MW8
"val_2"
IN1
OUT
%MW8
"val_2"
1
IN2
%M0.6
"decrease_2"
SUB
Int
EN
ENO
%MW8
"val_2"
IN1
OUT
%MW8
"val_2"
1
IN2
%MW8
"val_2"
<
Int
%MW12
"sp_2"
%M0.5
"increase_2"
S
%MW8
"val_2"
>=
Int
%MW12
"sp_2"
%M0.5
"increase_2"
%M0.5
"increase_2"
R
%MW8
"val_2"
>
Int
%MW12
"sp_2"
%M0.6
"decrease_2"
S

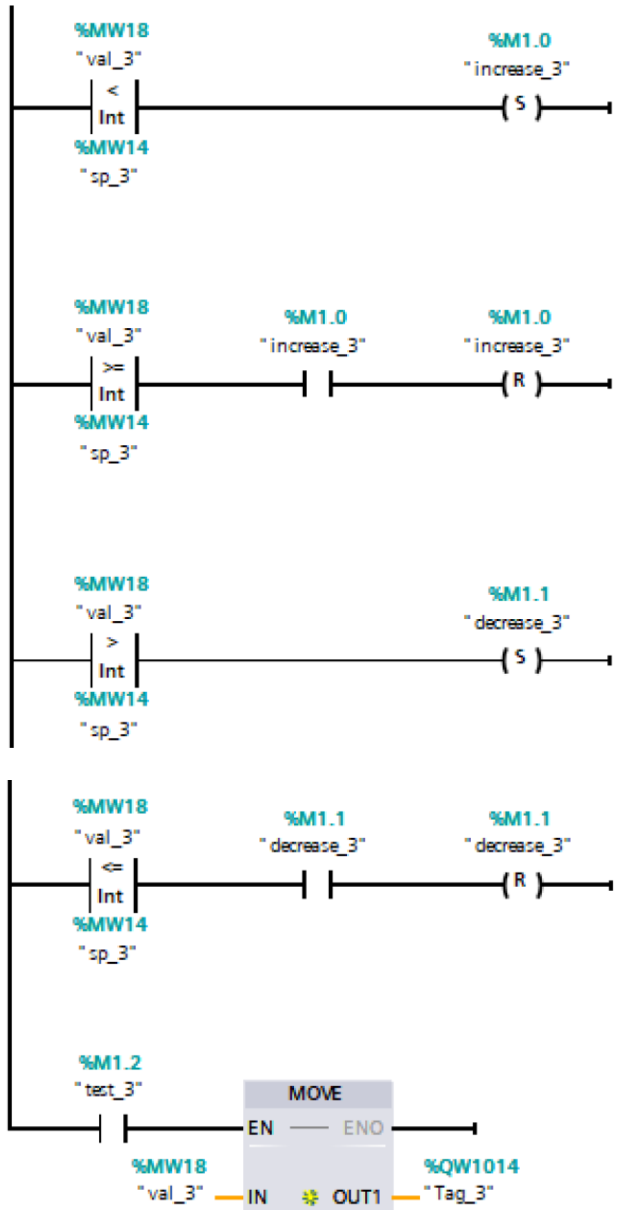


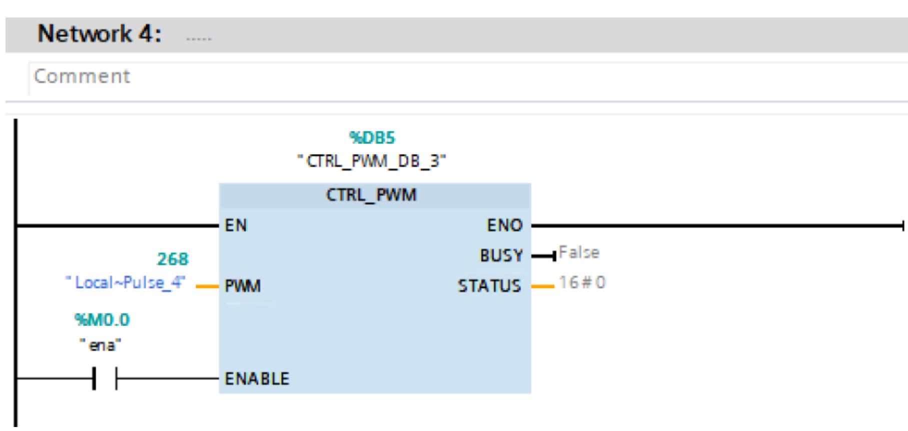

**Network 3:** .....

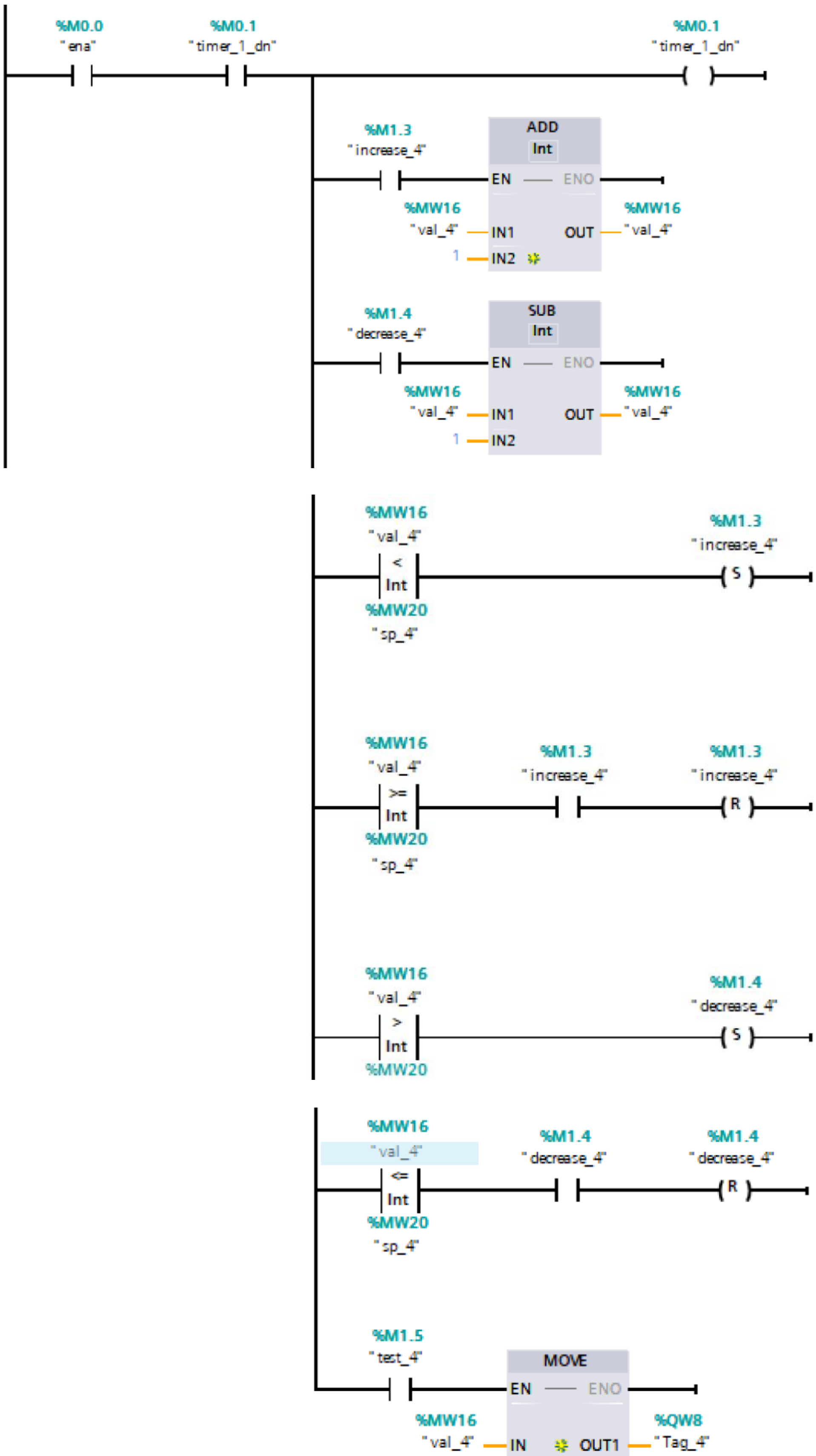
Comment

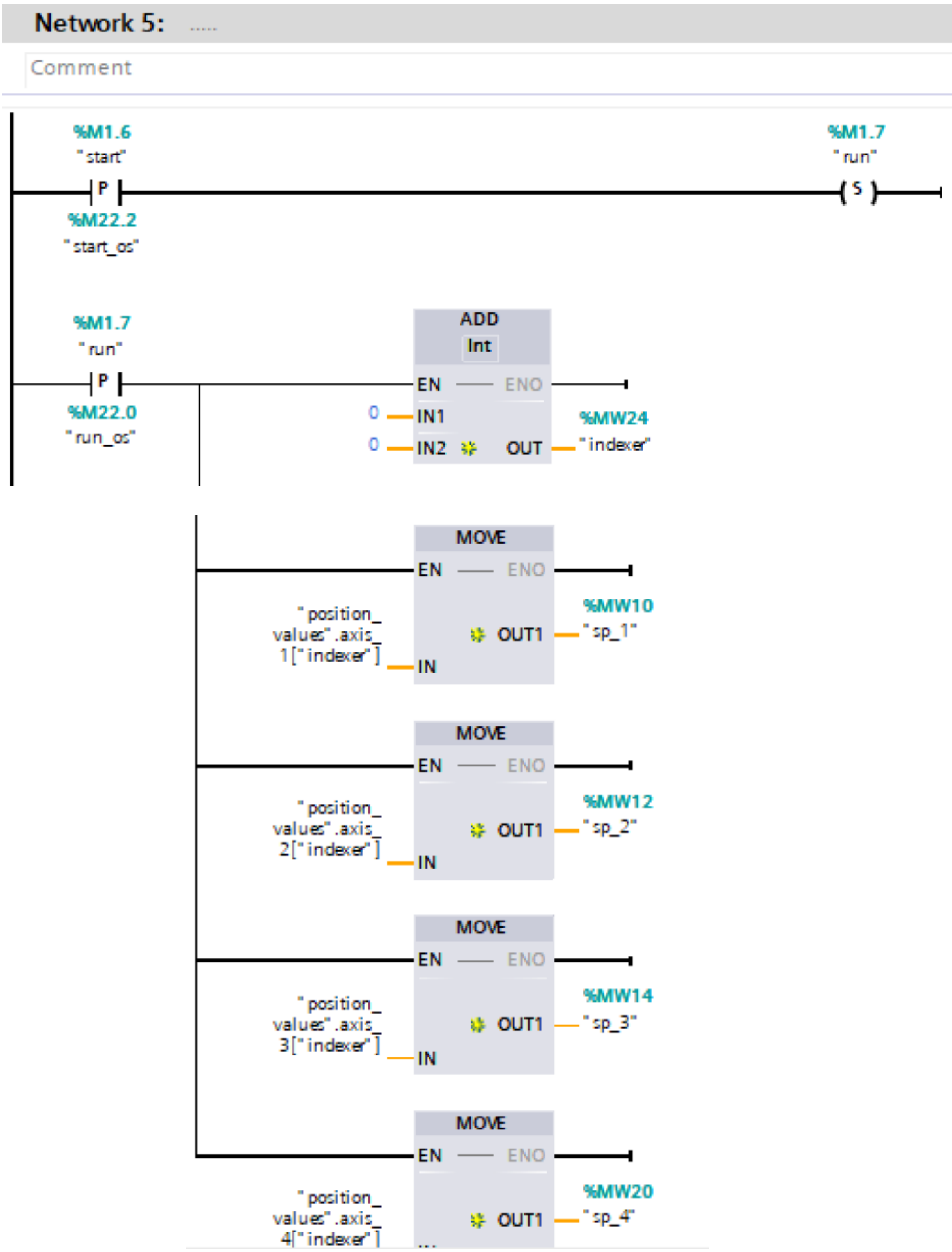


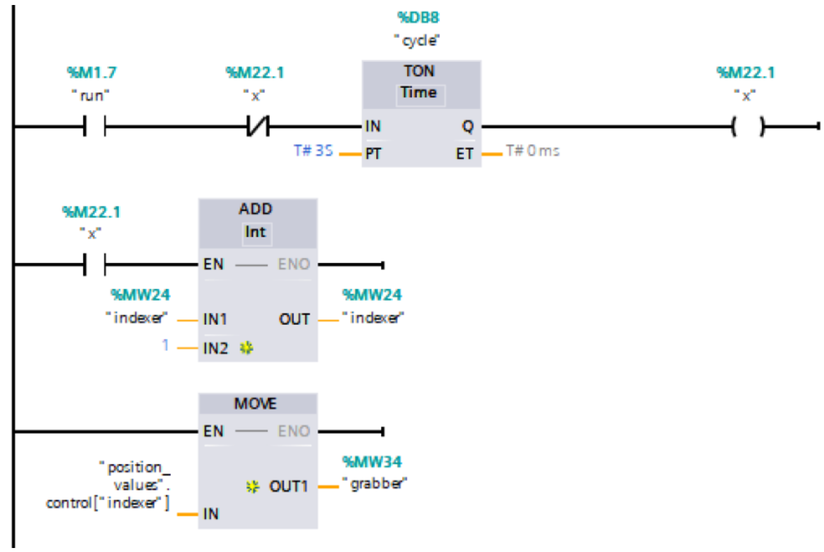



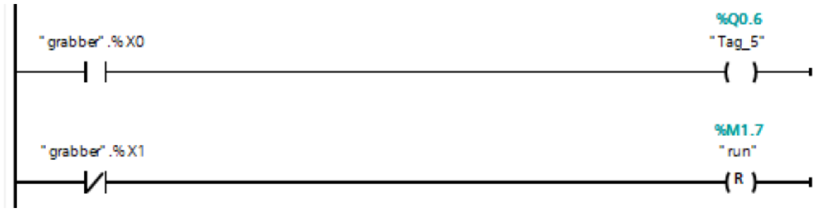

%MW18
"val_3"
<
Int
%MW14
"sp_3"
%M1.0
"increase_3"
S
%MW18
"val_3"
>=
Int
%MW14
"sp_3"
%M1.0
"increase_3"
%M1.0
"increase_3"
R
%MW18
"val_3"
>
Int
%MW14
"sp_3"
%M1.1
"decrease_3"
S
%MW18
"val_3"
<=
Int
%MW14
"sp_3"
%M1.1
"decrease_3"
%M1.1
"decrease_3"
R
%M1.2
"test_3"
MOVE
EN
ENO
%MW18
"val_3"
IN
OUT1
%QW1014
"Tag_3"

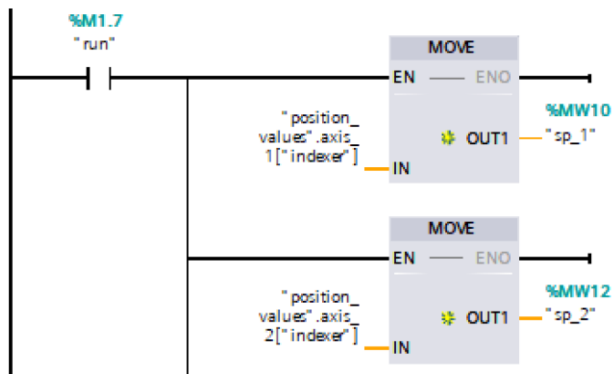

Network 4:
Comment
%DB5
"CTRL_PWM_DB_3"
CTRL_PWM
EN
ENO
268
"Local~Pulse_4"
PWM
BUSY
False
STATUS
16#0
%M0.0
"ena"
ENABLE

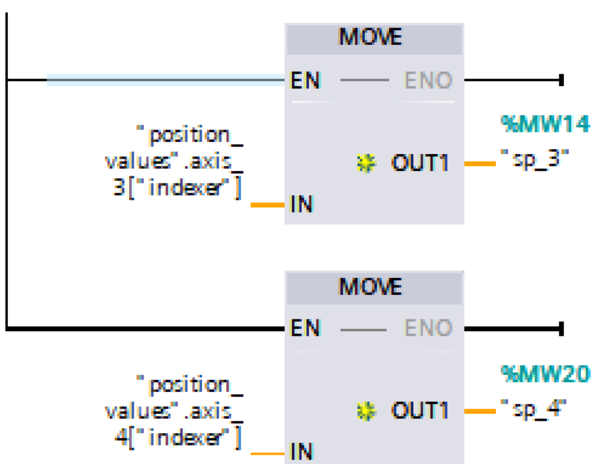

%M0.0
"ena"
%M0.1
"timer_1_dn"
%M0.1
"timer_1_dn"
%M1.3
"increase_4"
ADD
Int
EN
ENO
%MW16
"val_4"
IN1
OUT
%MW16
"val_4"
1
IN2
%M1.4
"decrease_4"
SUB
Int
EN
ENO
%MW16
"val_4"
IN1
OUT
%MW16
"val_4"
1
IN2
%MW16
"val_4"
<
Int
%MW20
"sp_4"
%M1.3
"increase_4"
S
%MW16
"val_4"
>=
Int
%MW20
"sp_4"
%M1.3
"increase_4"
%M1.3
"increase_4"
R
%MW16
"val_4"
>
Int
%MW20
%M1.4
"decrease_4"
S
%MW16
"val_4"
<=
Int
%MW20
"sp_4"
%M1.4
"decrease_4"
%M1.4
"decrease_4"
R
%M1.5
"test_4"
MOVE
EN
ENO
%MW16
"val_4"
IN
OUT1
%QW8
"Tag_4"

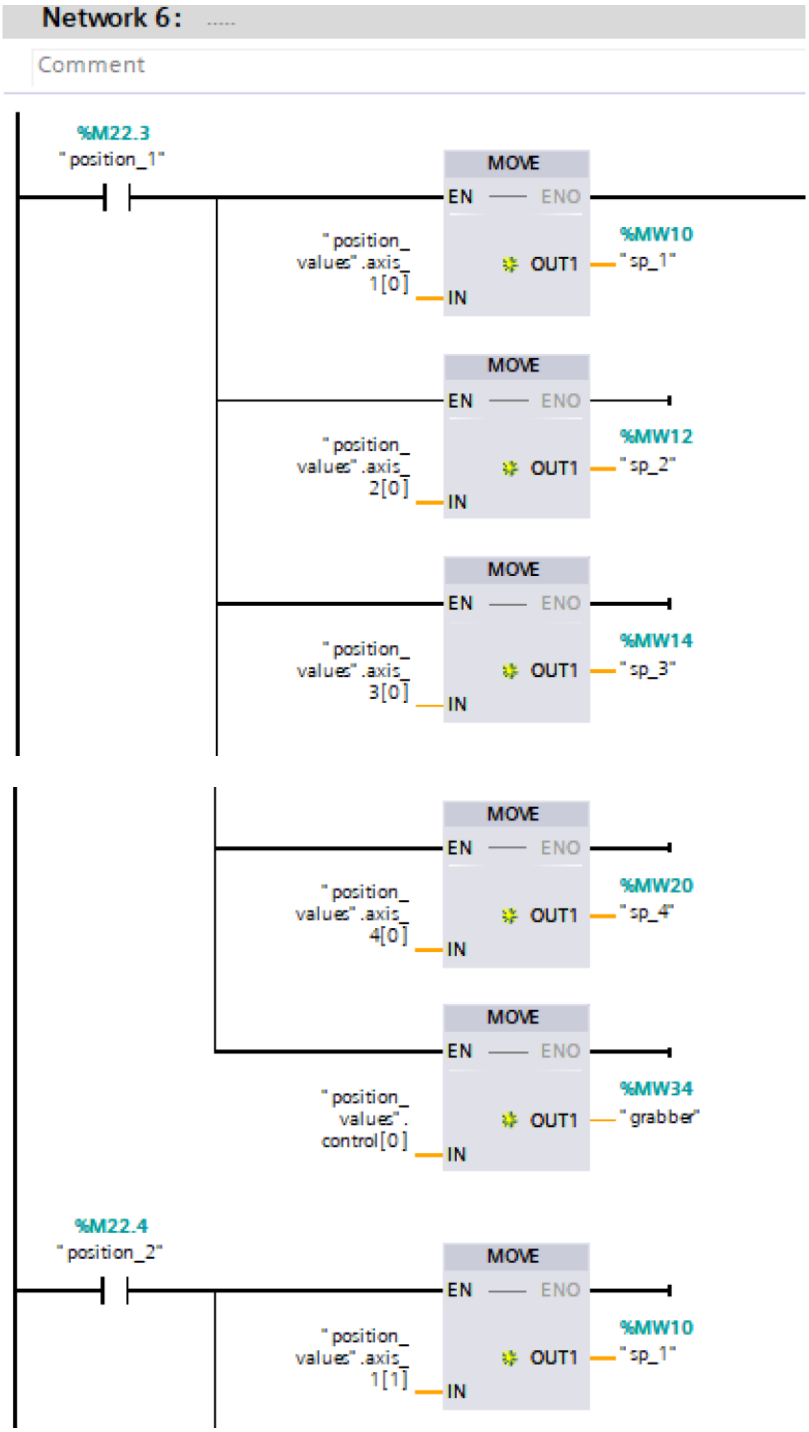

Network 5:
Comment
%M1.6
"start"
P
%M22.2
"start_os"
%M1.7
"run"
S
%M1.7
"run"
P
%M22.0
"run_os"
ADD
Int
EN
ENO
0
IN1
0
IN2
OUT
%MW24
"indexer"
MOVE
EN
ENO
"position_values".axis_1["indexer"]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_values".axis_2["indexer"]
IN
OUT1
%MW12
"sp_2"
MOVE
EN
ENO
"position_values".axis_3["indexer"]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_values".axis_4["indexer"]
OUT1
%MW20
"sp_4"

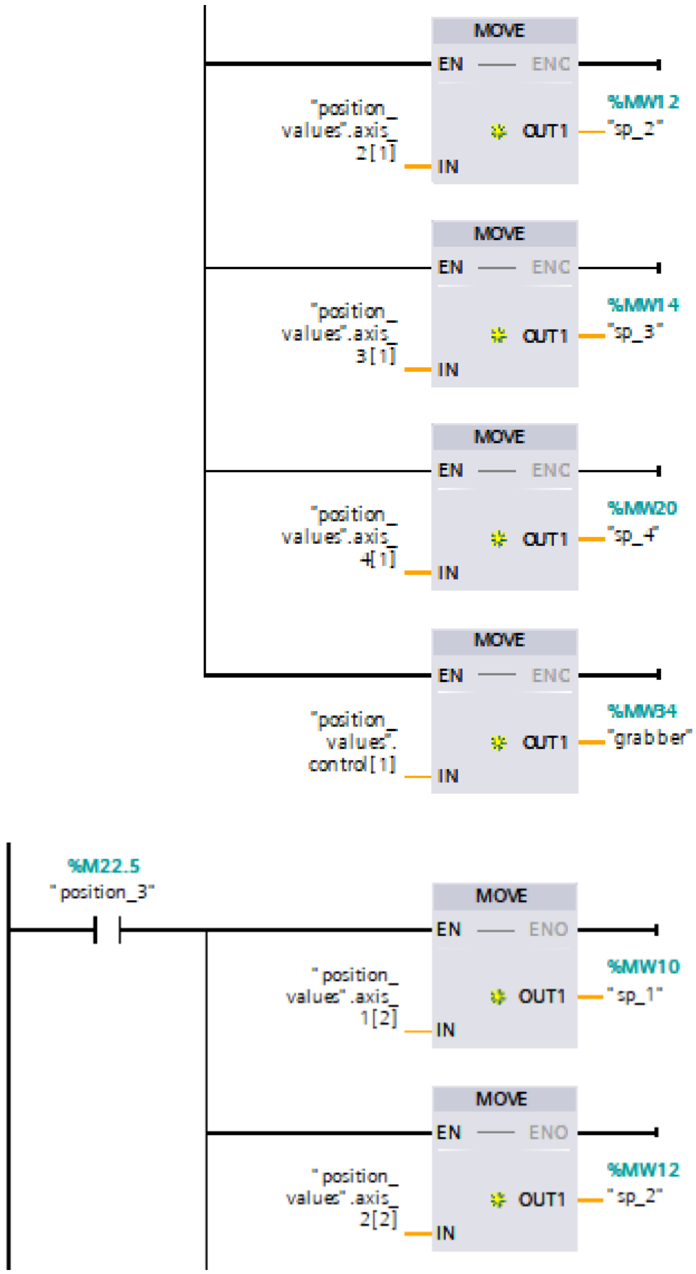

%DB8
"cycle"
TON
Time
%M1.7
"run"
%M22.1
"x"
IN
Q
T# 3S
PT
ET
T# 0 ms
%M22.1
"x"
%M22.1
"x"
ADD
Int
EN
ENO
%MW24
"indexer"
IN1
OUT
%MW24
"indexer"
1
IN2
MOVE
EN
ENO
"position_
values".
control["indexer"]
OUT1
%MW34
"grabber"
IN

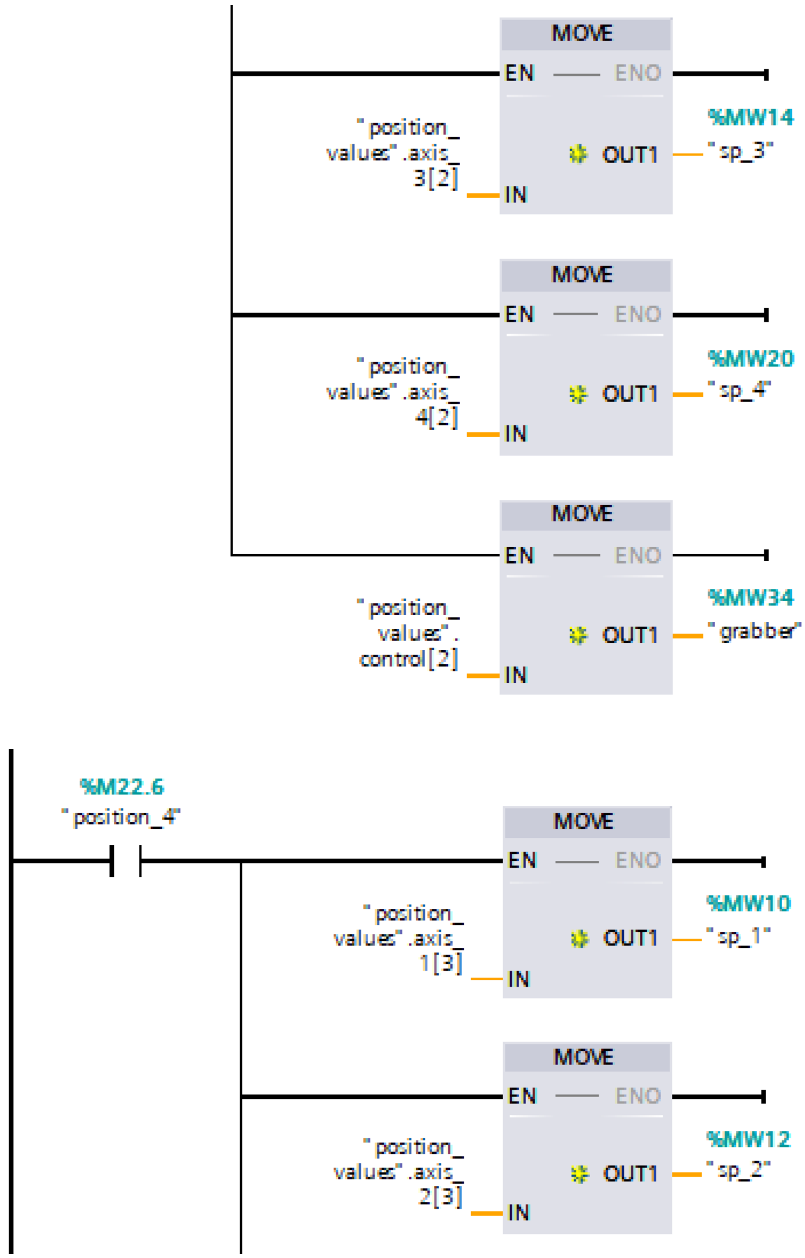

"grabber".%X0
%Q0.6
"Tag_5"
"grabber".%X1
%M1.7
"run"
R

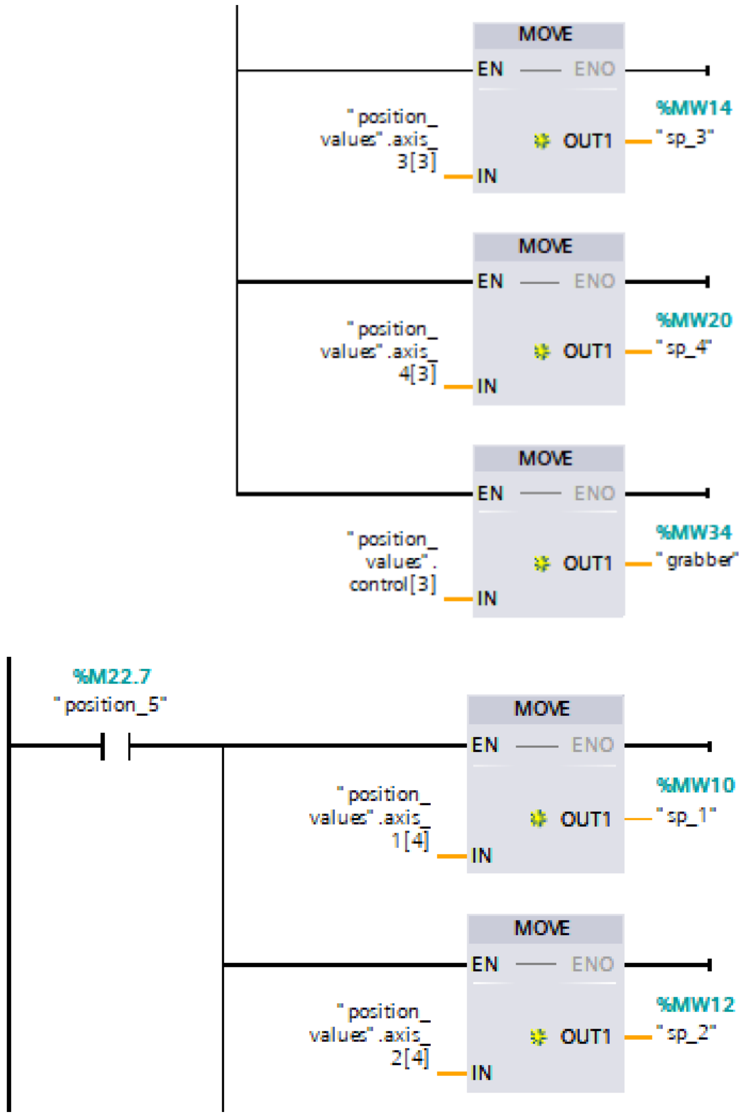

%M1.7
"run"
MOVE
EN
ENO
"position_
values".axis_
1["indexer"]
OUT1
%MW10
"sp_1"
IN
MOVE
EN
ENO
"position_
values".axis_
2["indexer"]
OUT1
%MW12
"sp_2"
IN

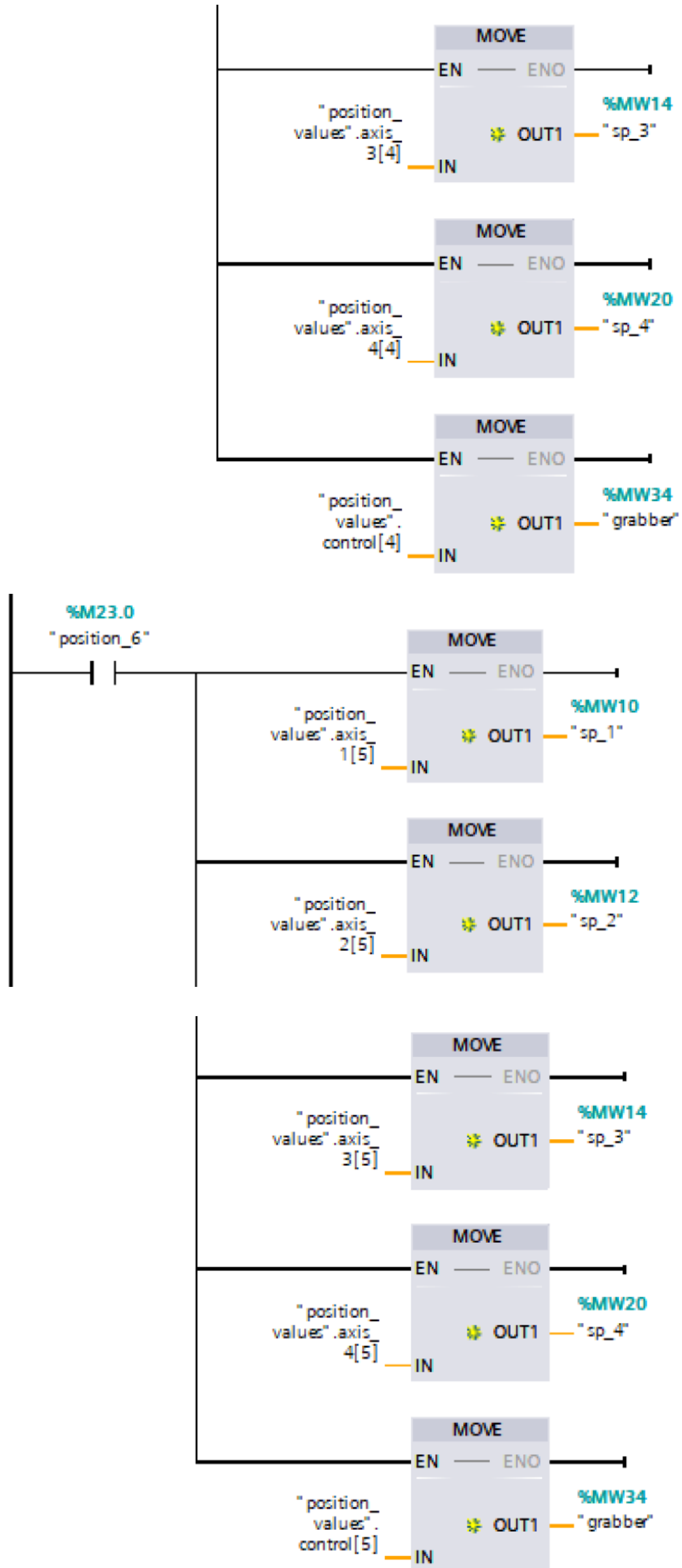

MOVE
EN
ENO
"position_
values".axis_
3["indexer"]
OUT1
%MW14
"sp_3"
IN
MOVE
EN
ENO
"position_
values".axis_
4["indexer"]
OUT1
%MW20
"sp_4"
IN

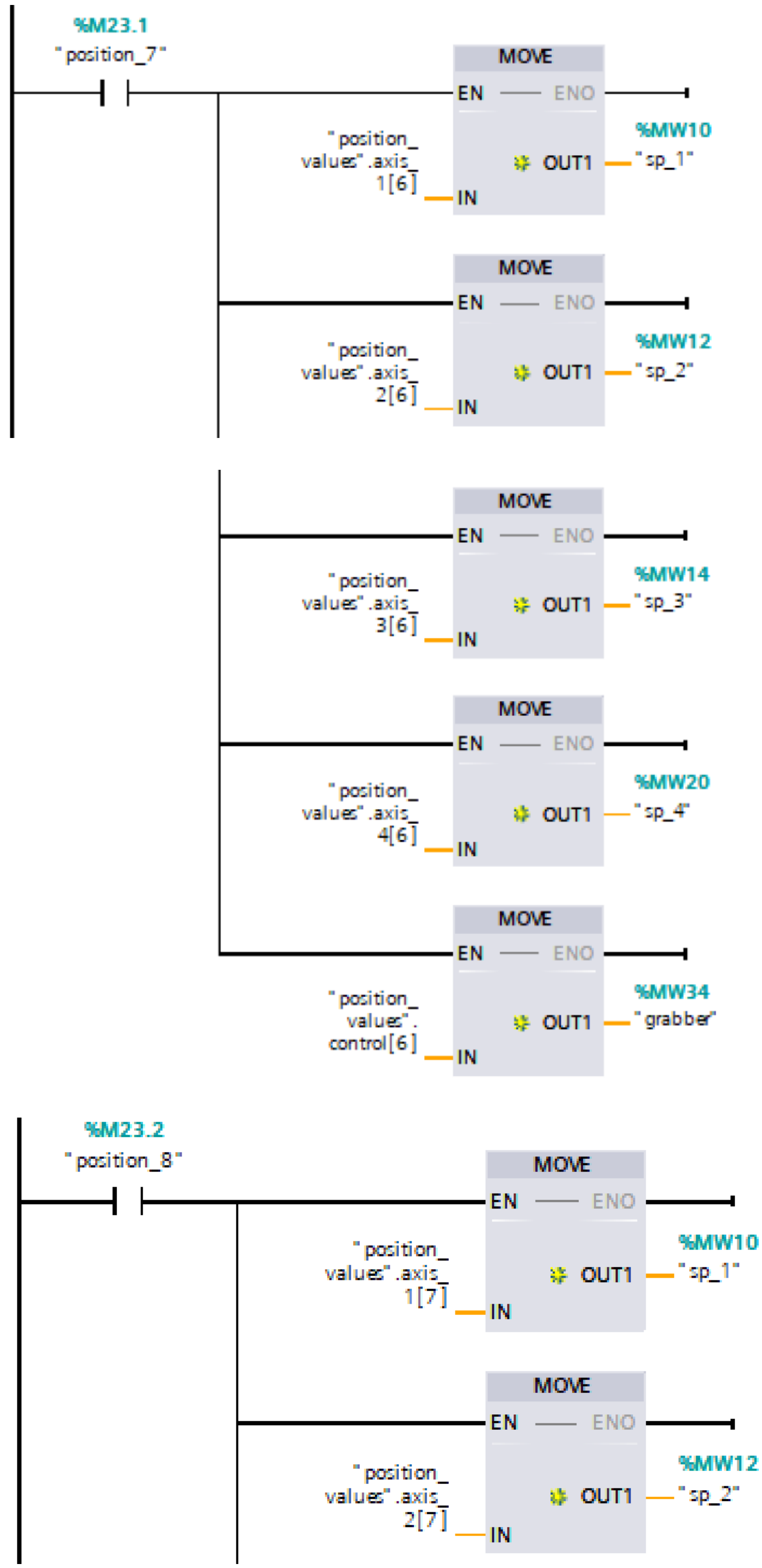

Network 6: .....
Comment
%M22.3
"position_1"
MOVE
EN
ENO
"position_values".axis_1[0]
IN
OUT1
%MW10
"sp_1"
"position_values".axis_2[0]
%MW12
"sp_2"
"position_values".axis_3[0]
%MW14
"sp_3"
"position_values".axis_4[0]
%MW20
"sp_4"
"position_values".control[0]
%MW34
"grabber"
%M22.4
"position_2"
"position_values".axis_1[1]
%MW10
"sp_1"

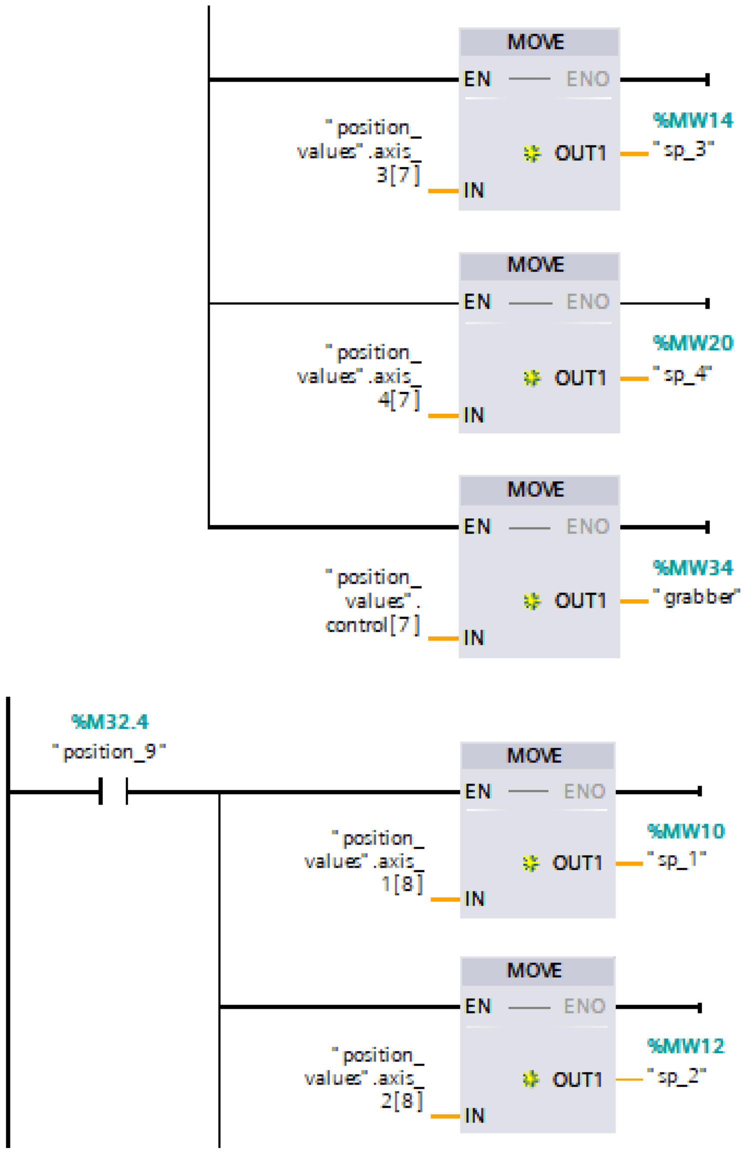

MOVE
EN
ENO
%MW12
"position_ values".axis_ 2[1]
OUT1
"sp_2"
IN
MOVE
EN
ENO
%MW14
"position_ values".axis_ 3[1]
OUT1
"sp_3"
IN
MOVE
EN
ENO
%MW20
"position_ values".axis_ 4[1]
OUT1
"sp_4"
IN
MOVE
EN
ENO
%MW34
"position_ values". control[1]
OUT1
"grabber"
IN
%M22.5
"position_3"
MOVE
EN
ENO
%MW10
"position_ values".axis_ 1[2]
OUT1
"sp_1"
IN
MOVE
EN
ENO
%MW12
"position_ values".axis_ 2[2]
OUT1
"sp_2"
IN

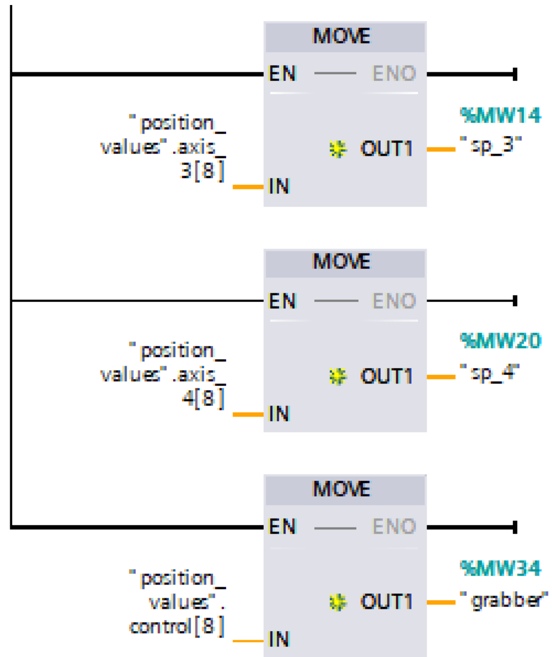

MOVE
EN
ENO
"position_ values".axis_ 3[2]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_ values".axis_ 4[2]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_ values". control[2]
IN
OUT1
%MW34
"grabber"
%M22.6
"position_4"
MOVE
EN
ENO
"position_ values".axis_ 1[3]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_ values".axis_ 2[3]
IN
OUT1
%MW12
"sp_2"


MOVE
EN
ENO
"position_ values".axis_ 3[3]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_ values".axis_ 4[3]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_ values". control[3]
IN
OUT1
%MW34
"grabber"
%M22.7
"position_5"
MOVE
EN
ENO
"position_ values".axis_ 1[4]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_ values".axis_ 2[4]
IN
OUT1
%MW12
"sp_2"


MOVE
EN
ENO
"position_values".axis_3[4]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_values".axis_4[4]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_values".control[4]
IN
OUT1
%MW34
"grabber"
%M23.0
"position_6"
MOVE
EN
ENO
"position_values".axis_1[5]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_values".axis_2[5]
IN
OUT1
%MW12
"sp_2"
MOVE
EN
ENO
"position_values".axis_3[5]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_values".axis_4[5]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_values".control[5]
IN
OUT1
%MW34
"grabber"

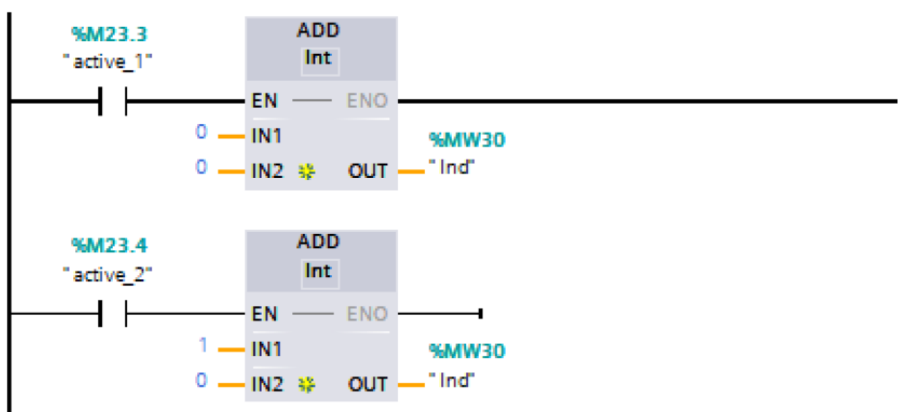

%M23.1
"position_7"
MOVE
EN
ENO
"position_values".axis_1[6]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_values".axis_2[6]
IN
OUT1
%MW12
"sp_2"
MOVE
EN
ENO
"position_values".axis_3[6]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_values".axis_4[6]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_values".control[6]
IN
OUT1
%MW34
"grabber"
%M23.2
"position_8"
MOVE
EN
ENO
"position_values".axis_1[7]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_values".axis_2[7]
IN
OUT1
%MW12
"sp_2"

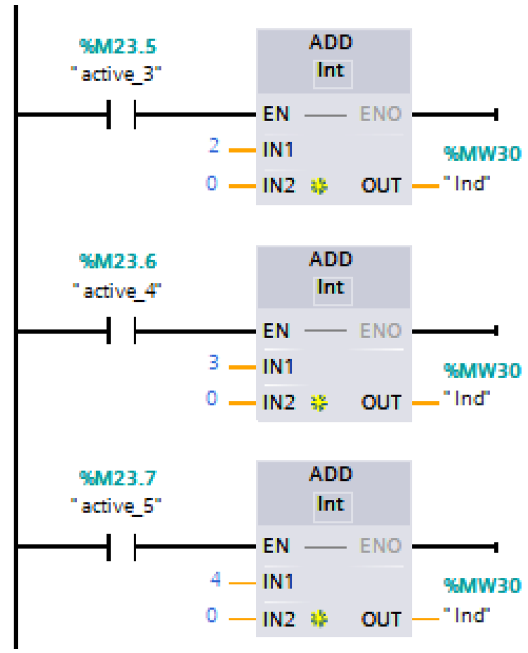

MOVE
EN
ENO
"position_values".axis_3[7]
IN
OUT1
%MW14
"sp_3"
MOVE
EN
ENO
"position_values".axis_4[7]
IN
OUT1
%MW20
"sp_4"
MOVE
EN
ENO
"position_values".control[7]
IN
OUT1
%MW34
"grabber"
%M32.4
"position_9"
MOVE
EN
ENO
"position_values".axis_1[8]
IN
OUT1
%MW10
"sp_1"
MOVE
EN
ENO
"position_values".axis_2[8]
IN
OUT1
%MW12
"sp_2"

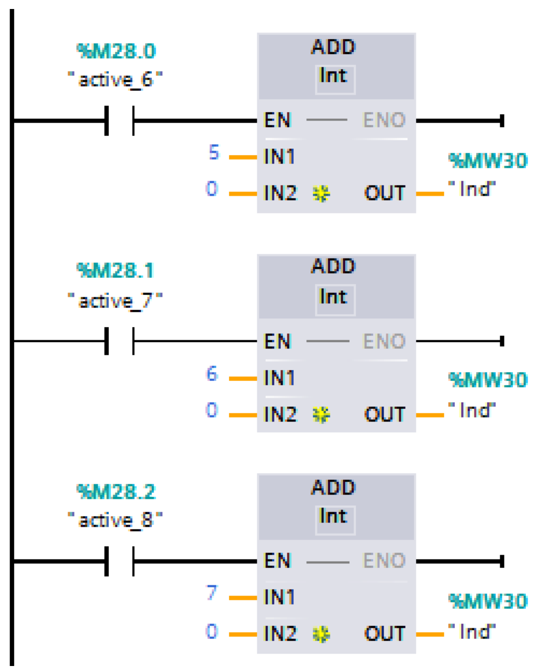


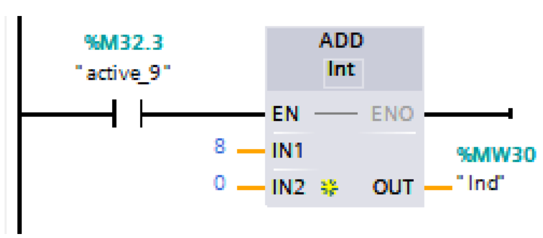

**Network 7:** .....

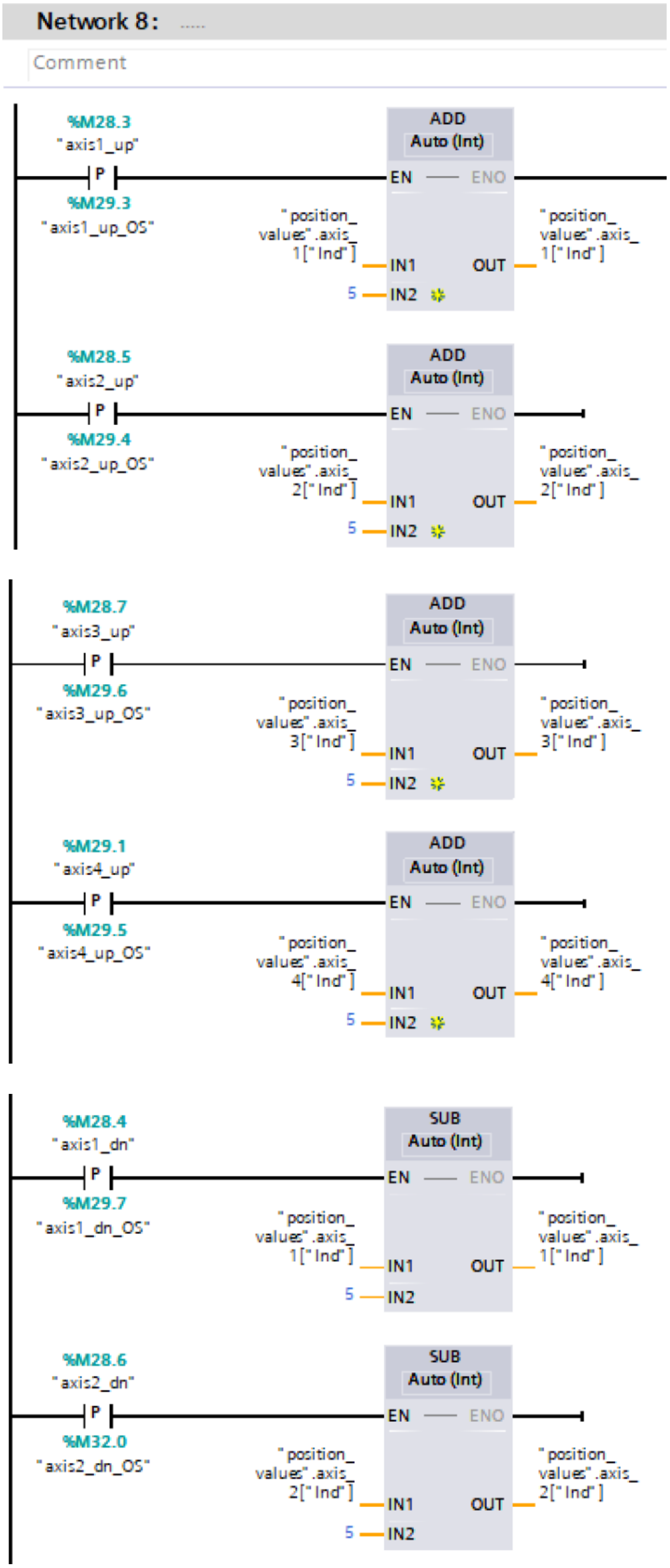
Comment

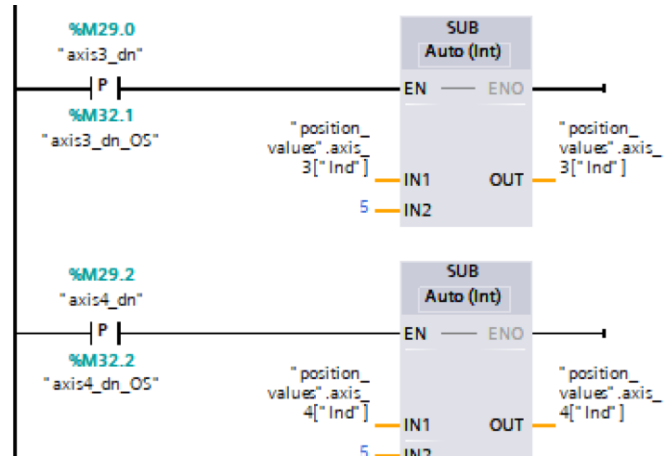


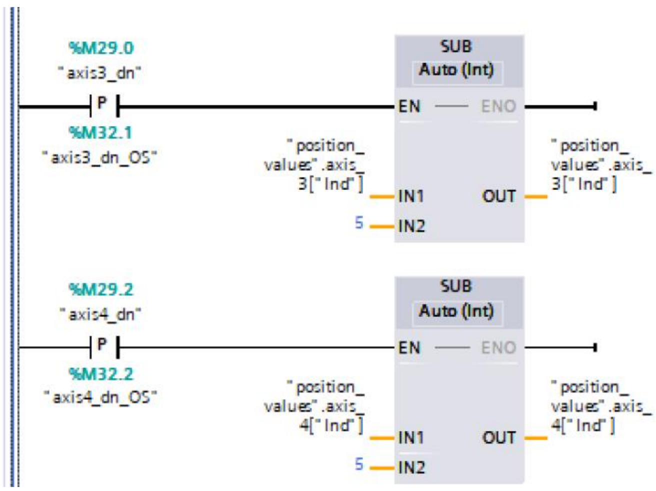

%M23.5
"active_3"
ADD
Int
EN
ENO
2
IN1
0
IN2
OUT
%MW30
"Ind"
%M23.6
"active_4"
ADD
Int
EN
ENO
3
IN1
0
IN2
OUT
%MW30
"Ind"
%M23.7
"active_5"
ADD
Int
EN
ENO
4
IN1
0
IN2
OUT
%MW30
"Ind"

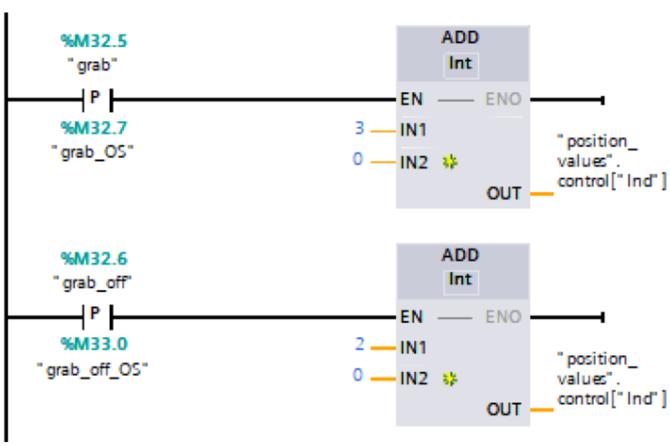

%M28.0
"active_6"
ADD
Int
EN
ENO
5
IN1
0
IN2
OUT
%MW30
"Ind"
%M28.1
"active_7"
ADD
Int
EN
ENO
6
IN1
0
IN2
OUT
%MW30
"Ind"
%M28.2
"active_8"
ADD
Int
EN
ENO
7
IN1
0
IN2
OUT
%MW30
"Ind"


%M32.3
"active_9"
ADD
Int
EN
ENO
8
IN1
0
IN2
OUT
%MW30
"Ind"


Network 8:
Comment
%M28.3
"axis1_up"
P
%M29.3
"axis1_up_OS"
ADD
Auto (Int)
EN
ENO
"position_values".axis_1["Ind"]
IN1
OUT
5
IN2
%M28.5
"axis2_up"
%M29.4
"axis2_up_OS"
"position_values".axis_2["Ind"]
%M28.7
"axis3_up"
%M29.6
"axis3_up_OS"
"position_values".axis_3["Ind"]
%M29.1
"axis4_up"
%M29.5
"axis4_up_OS"
"position_values".axis_4["Ind"]
%M28.4
"axis1_dn"
%M29.7
"axis1_dn_OS"
SUB
Auto (Int)
%M28.6
"axis2_dn"
%M32.0
"axis2_dn_OS"

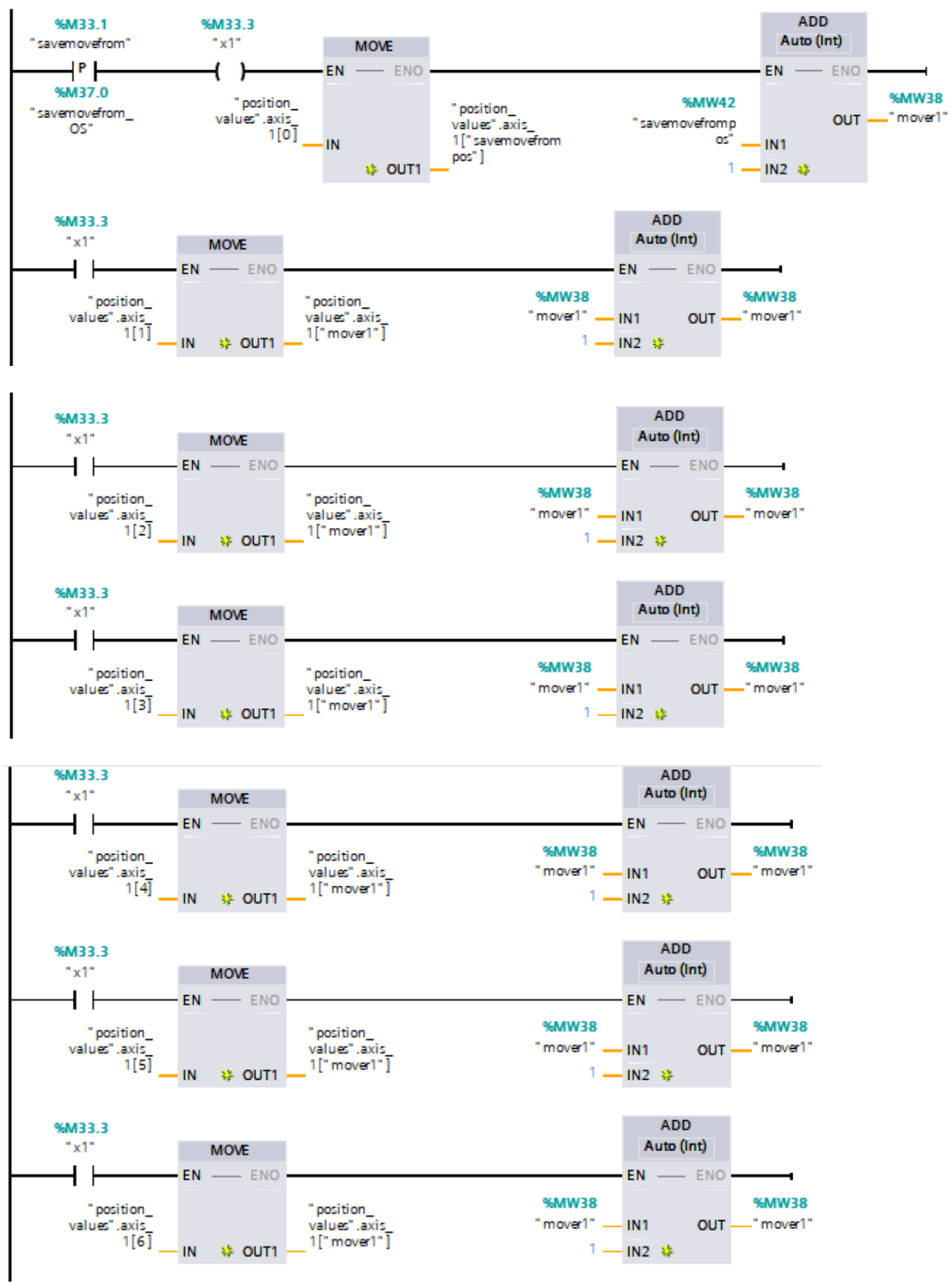

%M29.0
"axis3_dn"
P
%M32.1
"axis3_dn_OS"
SUB
Auto (Int)
EN
ENO
"position_values".axis_3["Ind"]
IN1
OUT
"position_values".axis_3["Ind"]
5
IN2
%M29.2
"axis4_dn"
P
%M32.2
"axis4_dn_OS"
SUB
Auto (Int)
EN
ENO
"position_values".axis_4["Ind"]
IN1
OUT
"position_values".axis_4["Ind"]
5

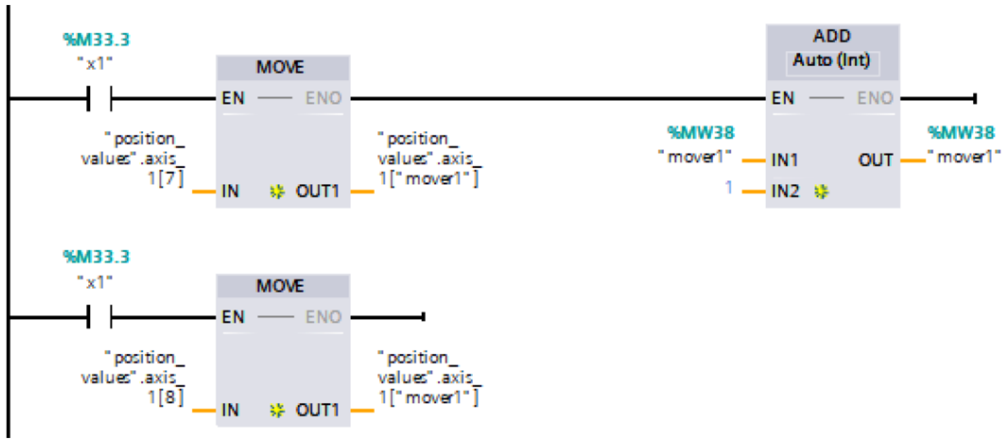

%M29.0
"axis3_dn"
P
%M32.1
"axis3_dn_OS"
SUB
Auto (Int)
EN
ENO
"position_values".axis_3["Ind"]
IN1
OUT
"position_values".axis_3["Ind"]
5
IN2
%M29.2
"axis4_dn"
P
%M32.2
"axis4_dn_OS"
SUB
Auto (Int)
EN
ENO
"position_values".axis_4["Ind"]
IN1
OUT
"position_values".axis_4["Ind"]
5
IN2

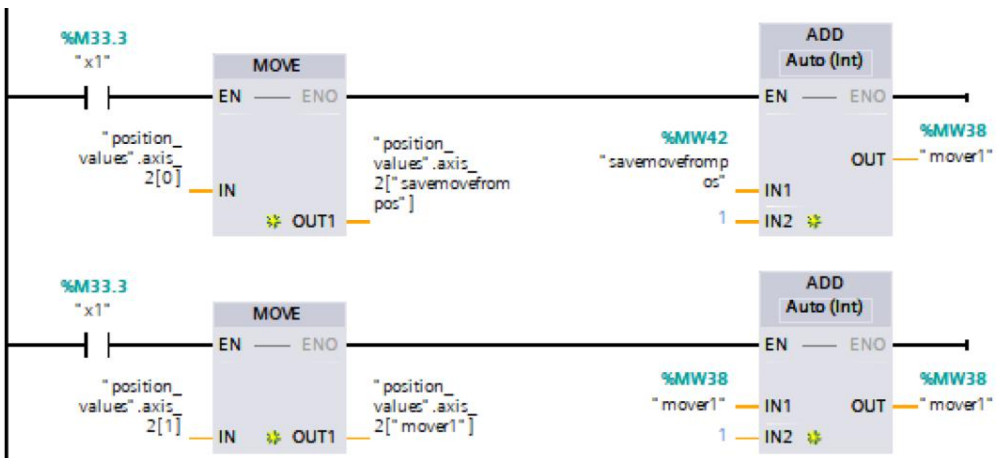

%M32.5
"grab"
P
%M32.7
"grab_OS"
ADD
Int
EN
ENO
3
IN1
0
IN2
OUT
"position_values".control["Ind"]
%M32.6
"grab_off"
P
%M33.0
"grab_off_OS"
ADD
Int
EN
ENO
2
IN1
0
IN2
OUT
"position_values".control["Ind"]

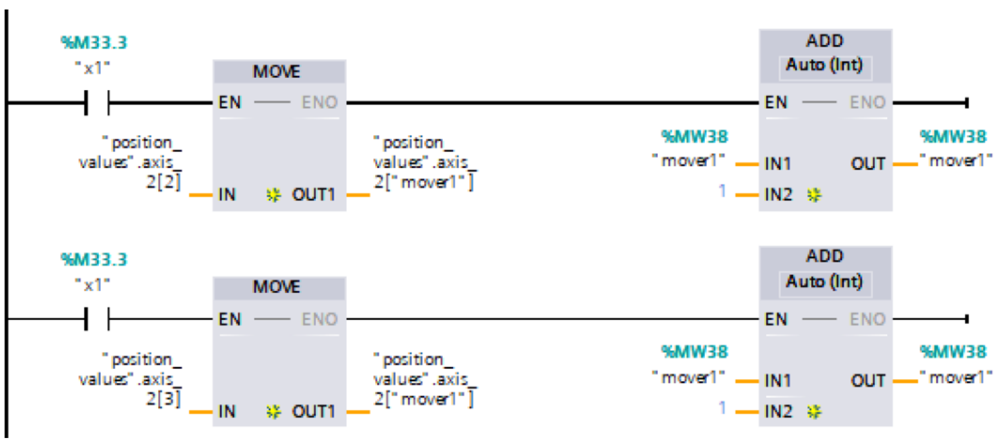
**Network 9:** .....

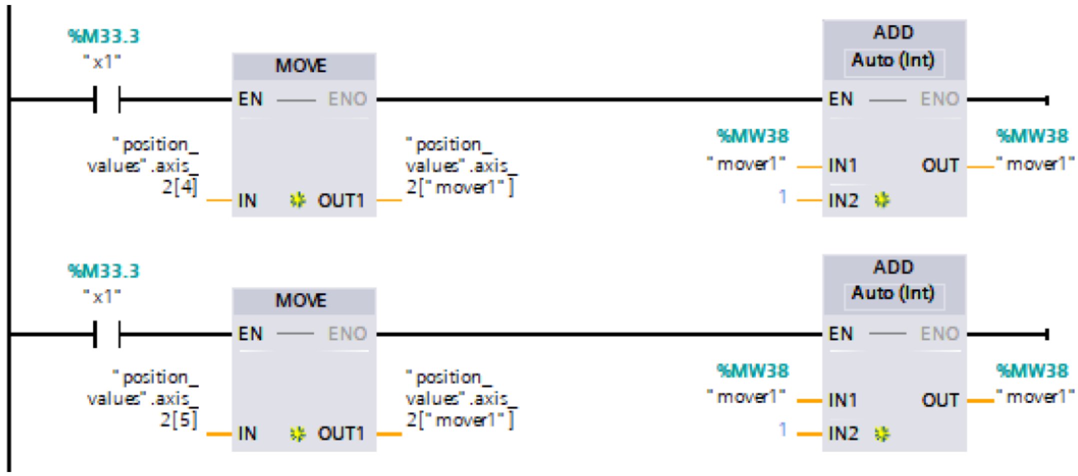
MOVER

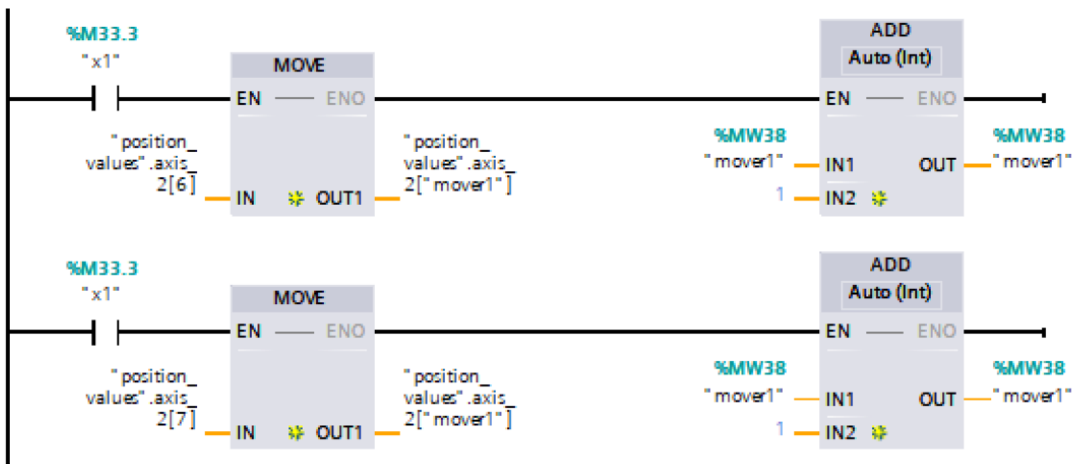


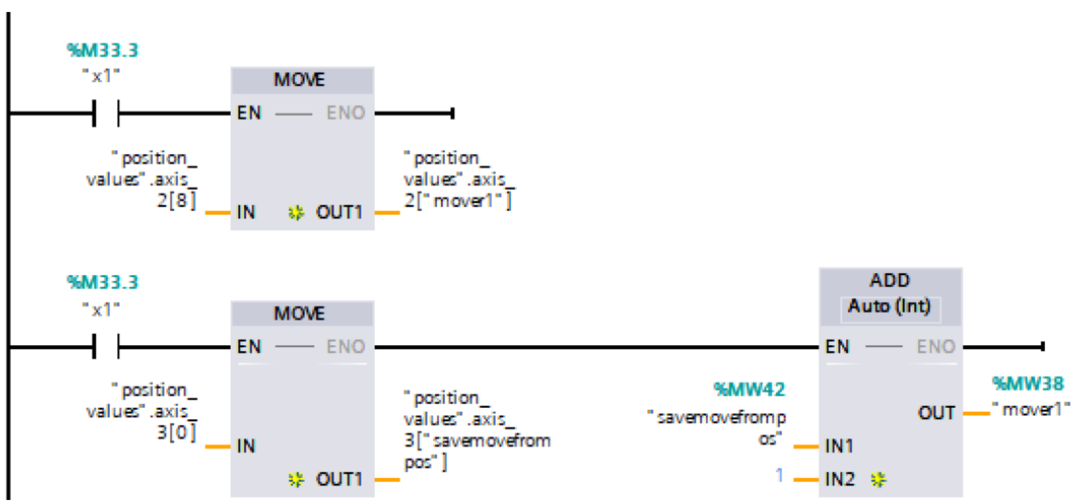

%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_1[7]
IN OUT1
"position_values".axis_1["mover1"]
ADD
Auto (Int)
%MW38
"mover1"
IN1
OUT
IN2
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
"position_values".axis_1[8]
"position_values".axis_1["mover1"]

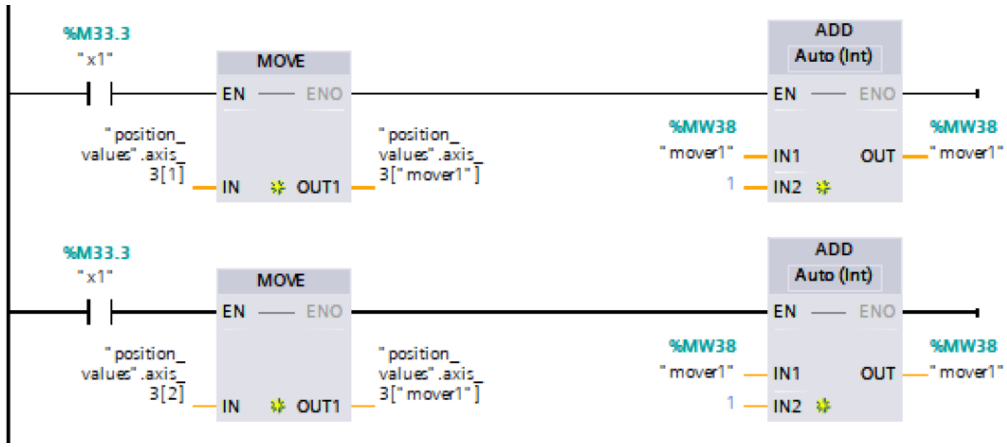

%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_2[0]
IN OUT1
"position_values".axis_2["savemovefrompos"]
ADD
Auto (Int)
%MW42
"savemovefrompos"
IN1
IN2
1
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
"position_values".axis_2[1]
"position_values".axis_2["mover1"]
ADD
Auto (Int)
%MW38
"mover1"
%MW38
"mover1"

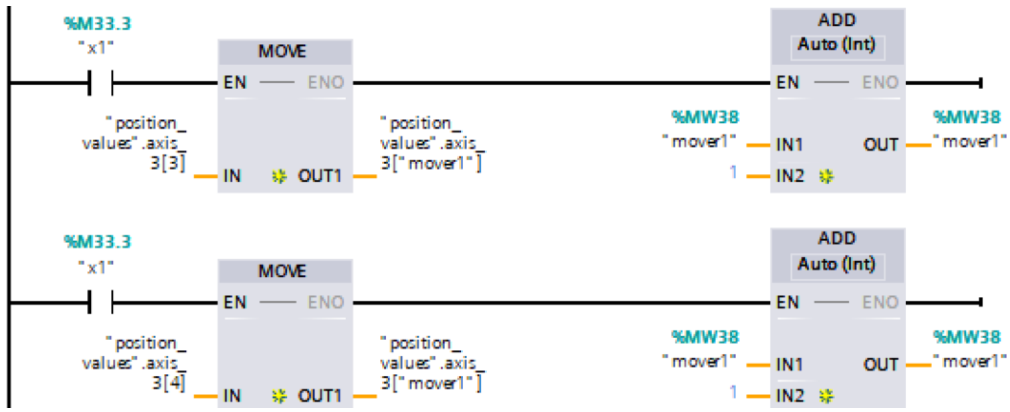

%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_2[2]
IN OUT1
"position_values".axis_2["mover1"]
ADD
Auto (Int)
%MW38
"mover1"
IN1
OUT
IN2
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
"position_values".axis_2[3]
"position_values".axis_2["mover1"]
ADD
Auto (Int)
%MW38
"mover1"
%MW38
"mover1"

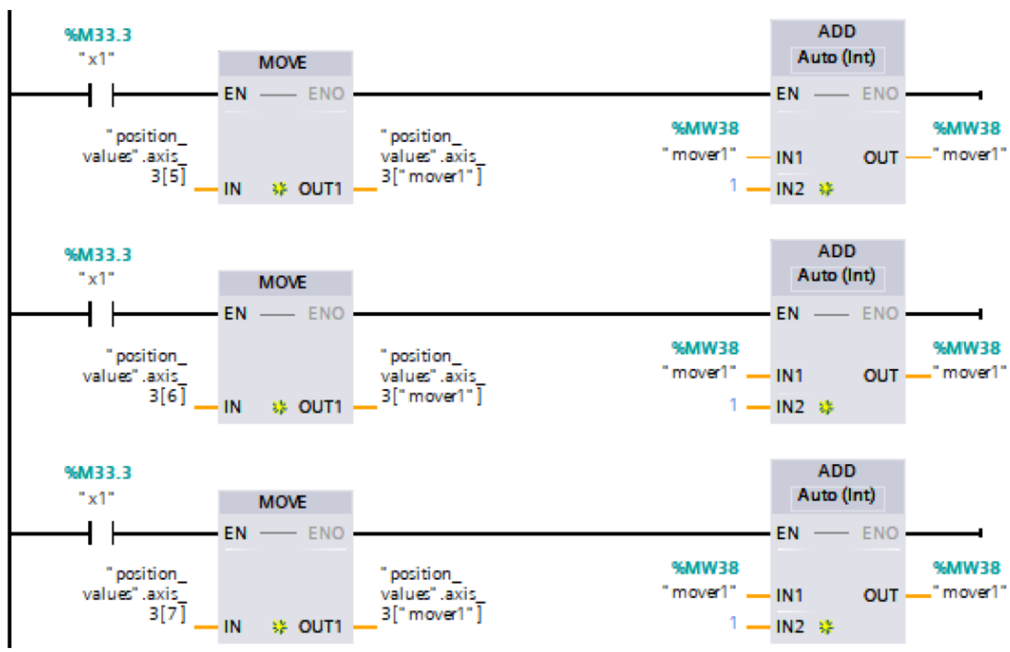

%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 2[4]
IN OUT1
"position_ values".axis_ 2["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 2[5]
IN OUT1
"position_ values".axis_ 2["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"

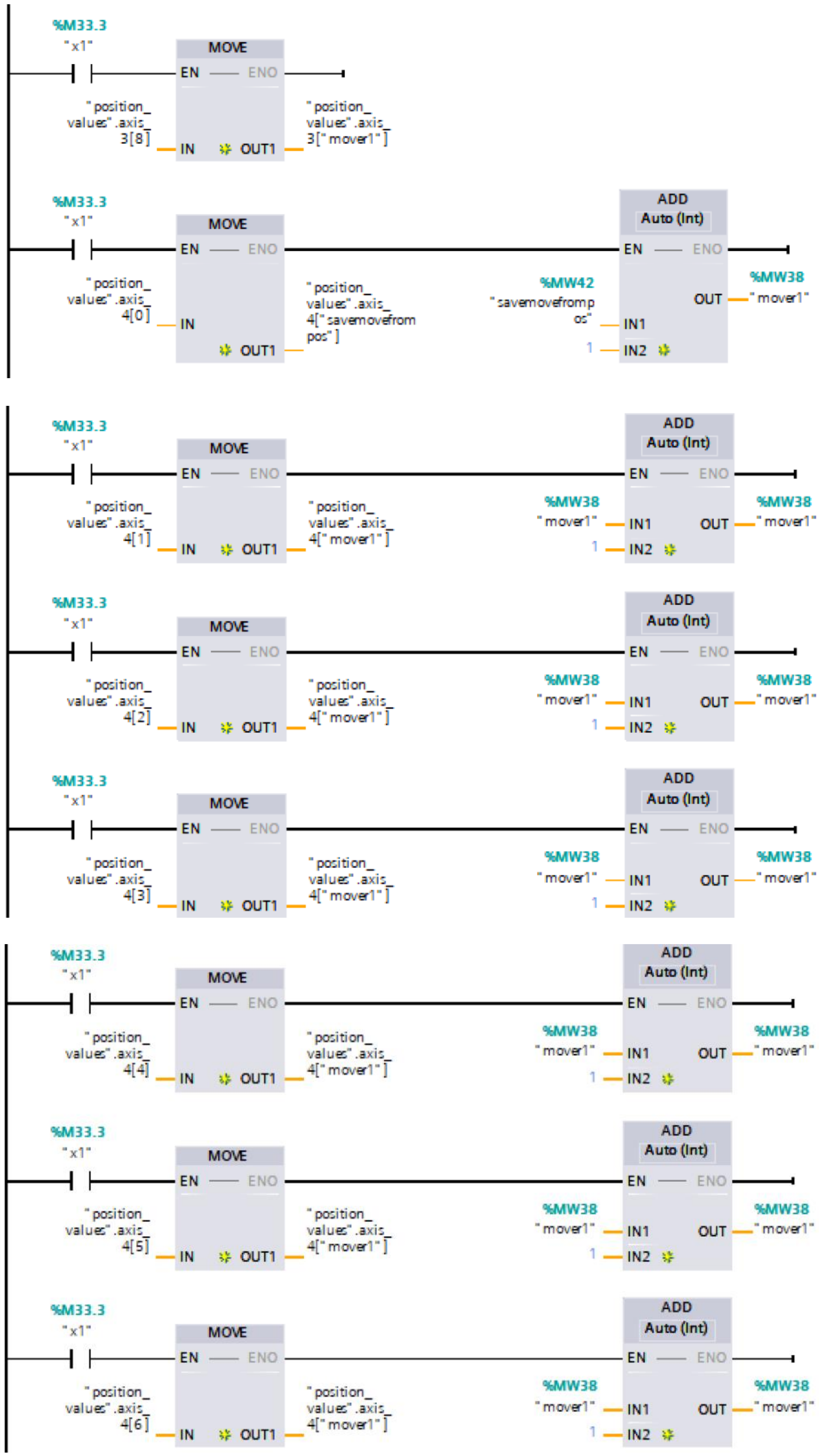

%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 2[6]
IN OUT1
"position_ values".axis_ 2["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 2[7]
IN OUT1
"position_ values".axis_ 2["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"

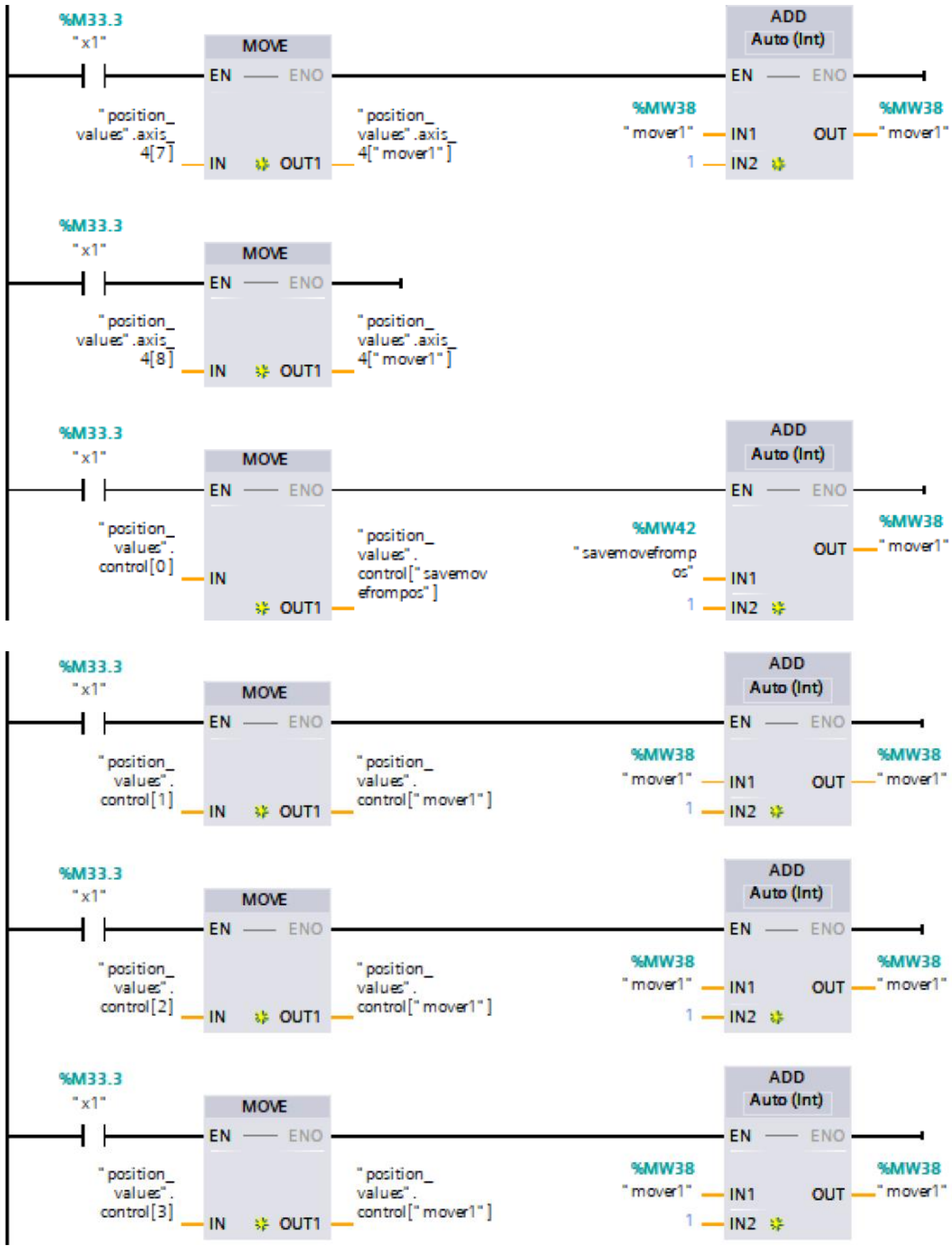

%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 2[8]
IN OUT1
"position_ values".axis_ 2["mover1"]
%M33.3
"x1"
MOVE
EN ENO
"position_ values".axis_ 3[0]
IN OUT1
"position_ values".axis_ 3["savemovefrom pos"]
ADD
Auto (Int)
EN ENO
%MW42
"savemovefromp os"
IN1 OUT
1
IN2
%MW38
"mover1"

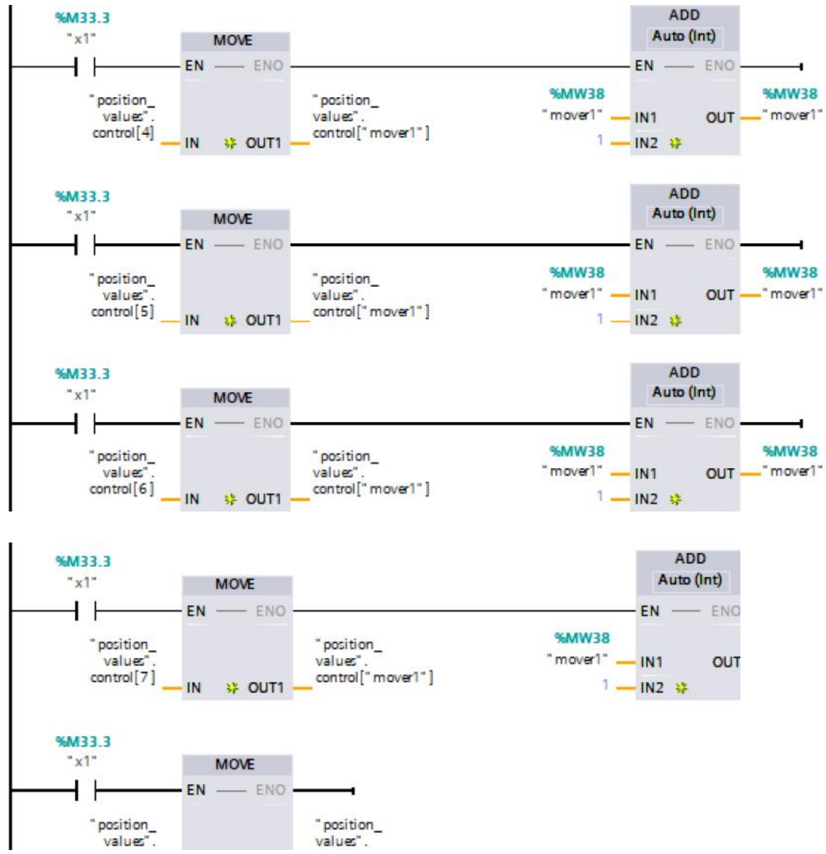

%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[1]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2
%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[2]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2


%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[3]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2
%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[4]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2


%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[5]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2
%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[6]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2
%M33.3
"x1"
MOVE
EN
ENO
"position_values".axis_3[7]
IN
OUT1
"position_values".axis_3["mover1"]
ADD
Auto (Int)
EN
ENO
%MW38
"mover1"
IN1
OUT
%MW38
"mover1"
1
IN2

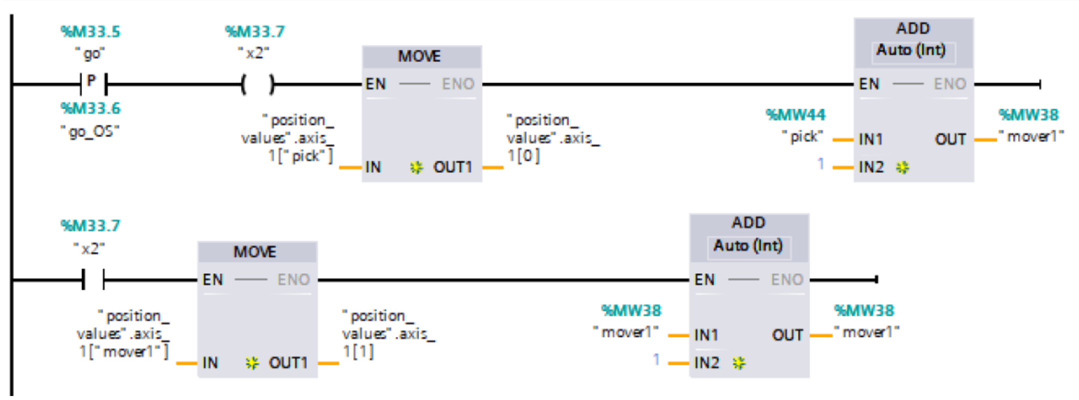

%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_3[8]
IN OUT1
"position_values".axis_3["mover1"]
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[0]
IN OUT1
"position_values".axis_4["savemovefrompos"]
ADD
Auto (Int)
EN ENO
%MW42
"savemovefrompos"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[1]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[2]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[3]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[4]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[5]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
"position_values".axis_4[6]
IN OUT1
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1
1
IN2
OUT
%MW38
"mover1"

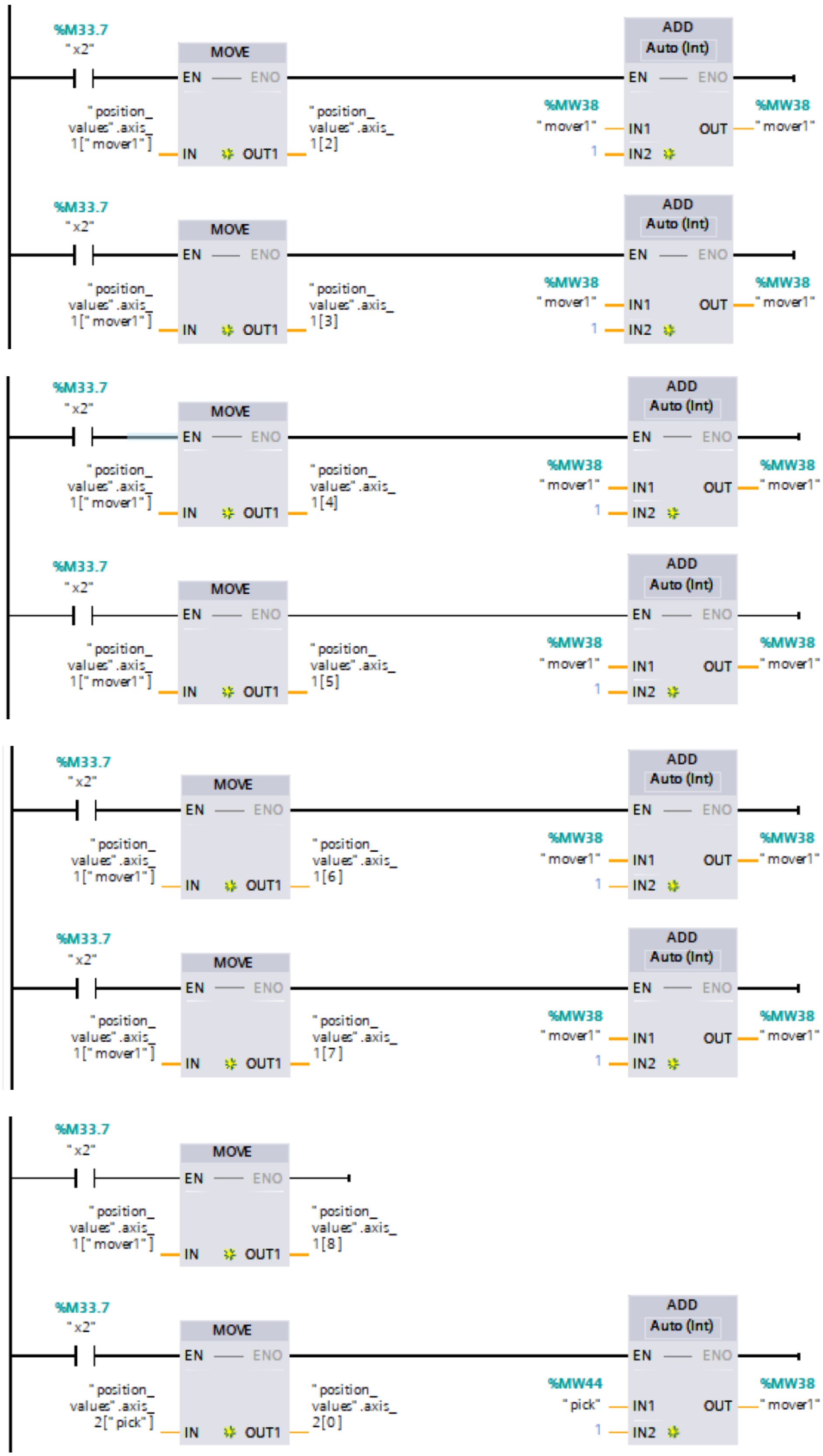

%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".axis_4[7]
"position_values".axis_4["mover1"]
ADD
Auto (Int)
EN ENO
IN1 OUT
IN2
%MW38
"mover1"
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".axis_4[8]
"position_values".axis_4["mover1"]
%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".control[0]
"position_values".control["savemovefrompos"]
ADD
Auto (Int)
EN ENO
IN1 OUT
IN2
%MW42
"savemovefrompos"
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".control[1]
"position_values".control["mover1"]
ADD
Auto (Int)
EN ENO
IN1 OUT
IN2
%MW38
"mover1"
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".control[2]
"position_values".control["mover1"]
ADD
Auto (Int)
EN ENO
IN1 OUT
IN2
%MW38
"mover1"
1
%MW38
"mover1"
%M33.3
"x1"
MOVE
EN ENO
IN OUT1
"position_values".control[3]
"position_values".control["mover1"]
ADD
Auto (Int)
EN ENO
IN1 OUT
IN2
%MW38
"mover1"
1
%MW38
"mover1"

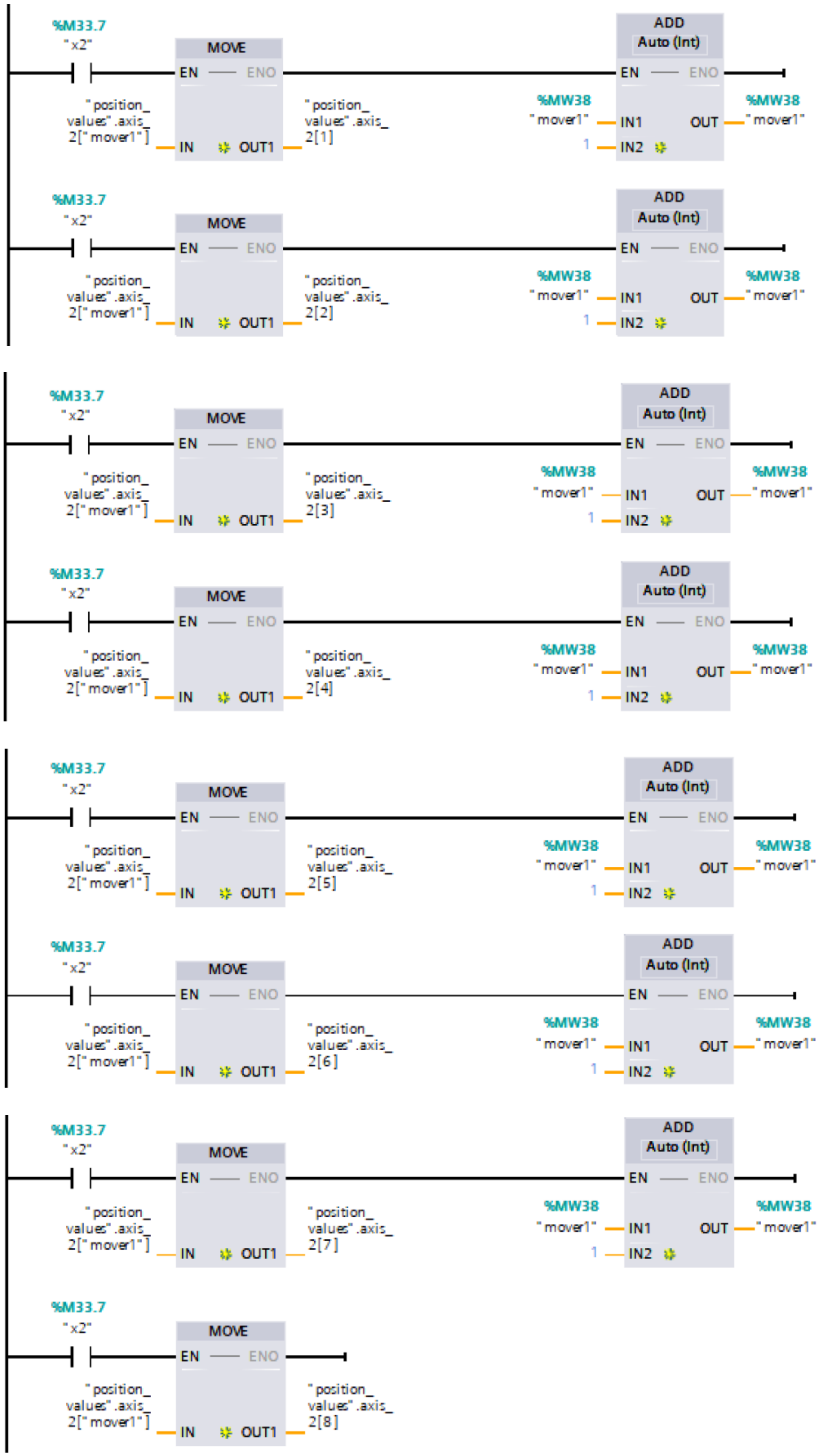


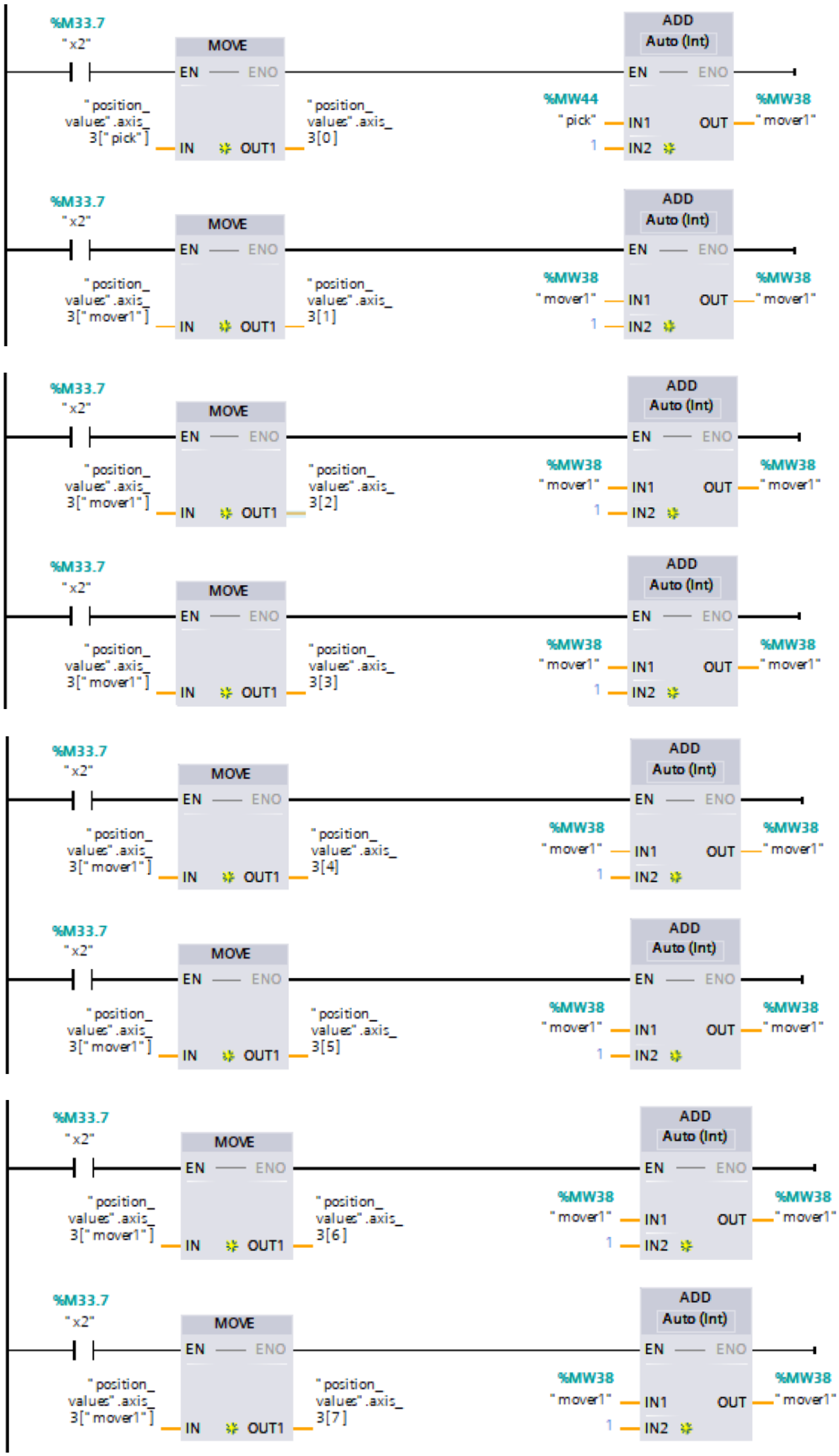

**Network 10:** .....

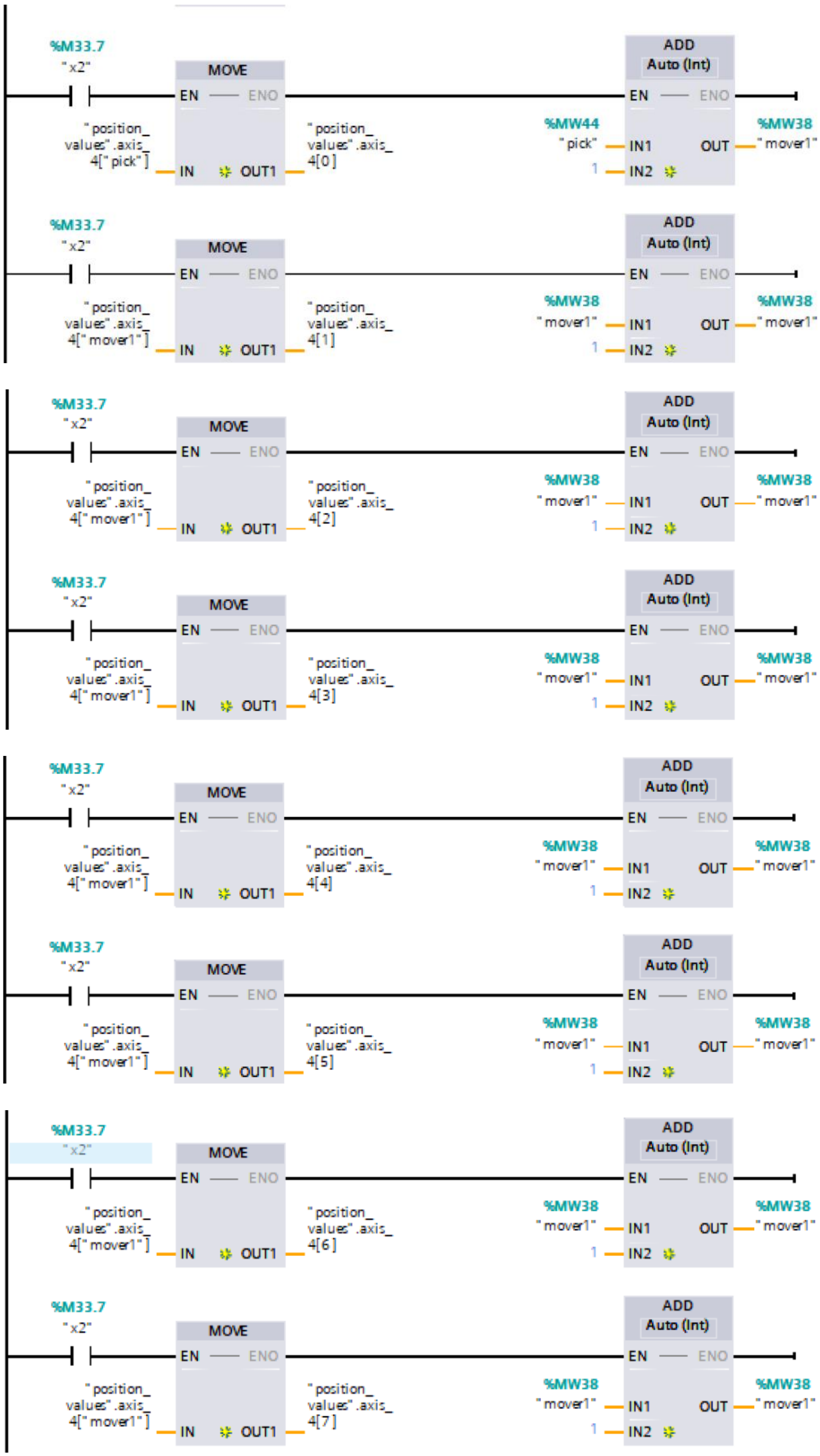
MOVER

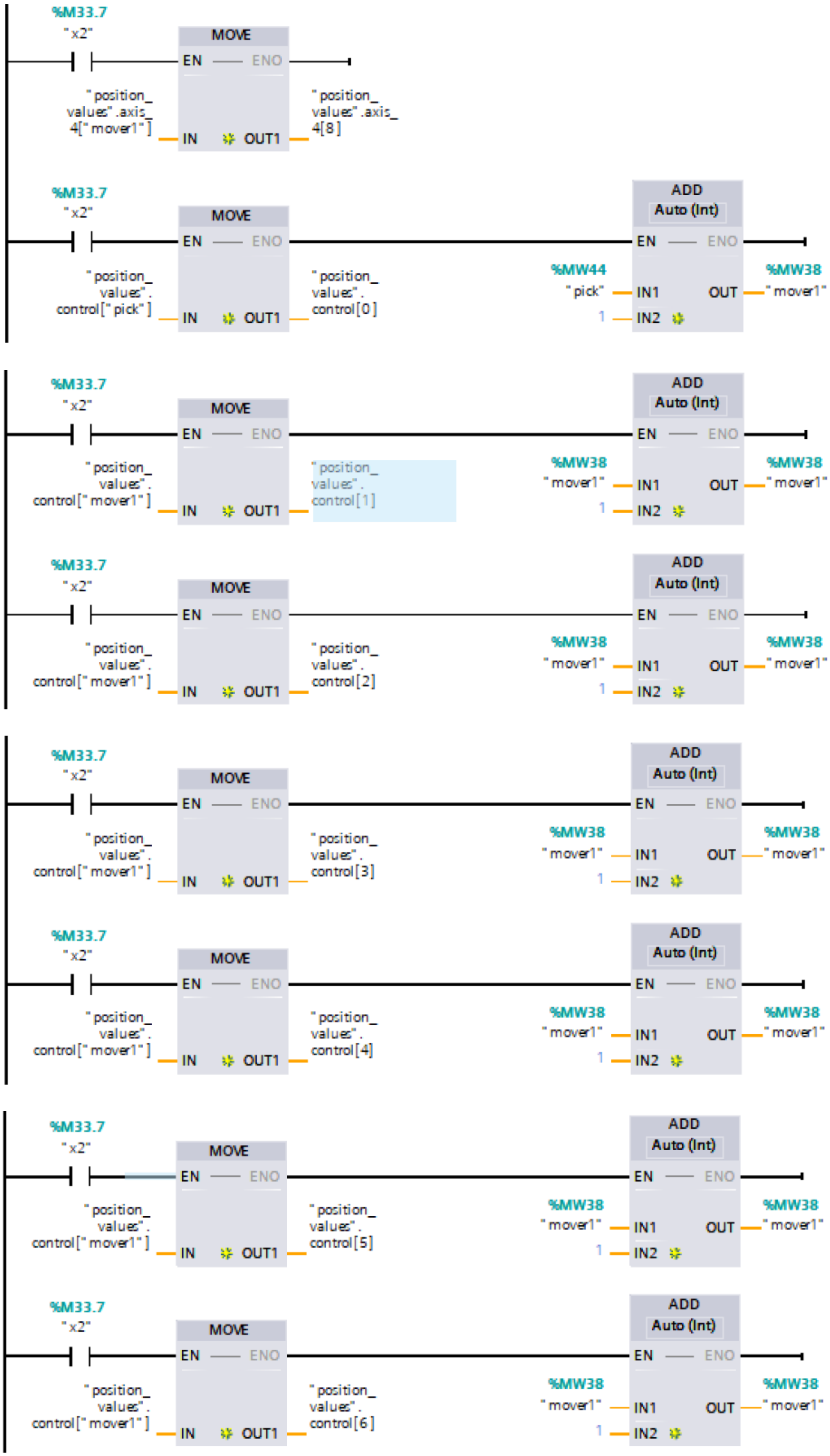


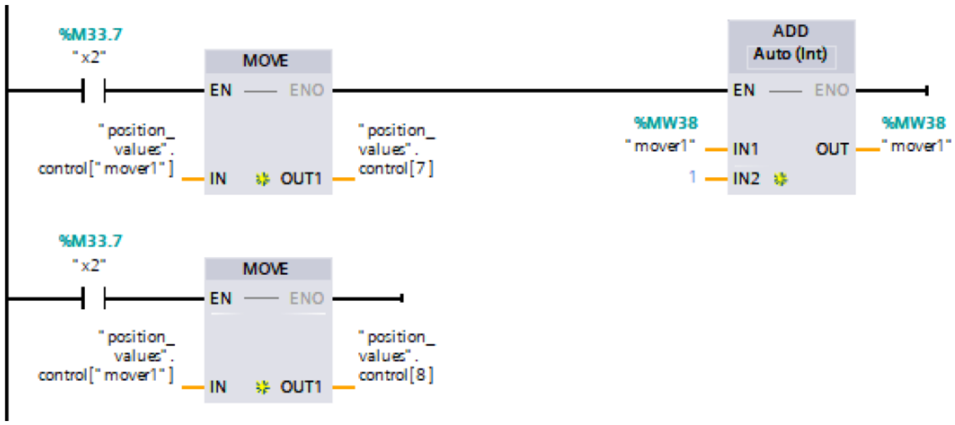

%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_1["mover1"]
IN OUT1
"position_values".axis_1[2]
ADD
Auto (Int)
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
"position_values".axis_1[3]
"position_values".axis_1[4]
"position_values".axis_1[5]
"position_values".axis_1[6]
"position_values".axis_1[7]
"position_values".axis_1[8]
"position_values".axis_2["pick"]
"position_values".axis_2[0]
%MW44
"pick"


%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[1]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[2]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[3]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[4]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[5]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[6]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[7]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_2["mover1"]
IN OUT1
"position_values".axis_2[8]


%M33.7
"x2"
MOVE
EN
ENO
"position_ values".axis_ 3["pick"]
IN
OUT1
"position_ values".axis_ 3[0]
ADD
Auto (Int)
%MW44
"pick"
IN1
IN2
1
OUT
%MW38
"mover1"
"position_ values".axis_ 3["mover1"]
"position_ values".axis_ 3[1]
"position_ values".axis_ 3[2]
"position_ values".axis_ 3[3]
"position_ values".axis_ 3[4]
"position_ values".axis_ 3[5]
"position_ values".axis_ 3[6]
"position_ values".axis_ 3[7]


%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["pick"]
IN OUT1
"position_values".axis_4[0]
ADD
Auto (Int)
EN ENO
%MW44
"pick"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[1]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[2]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[3]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[4]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[5]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[6]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"
%M33.7
"x2"
MOVE
EN ENO
"position_values".axis_4["mover1"]
IN OUT1
"position_values".axis_4[7]
ADD
Auto (Int)
EN ENO
%MW38
"mover1"
IN1 OUT
1
IN2
%MW38
"mover1"

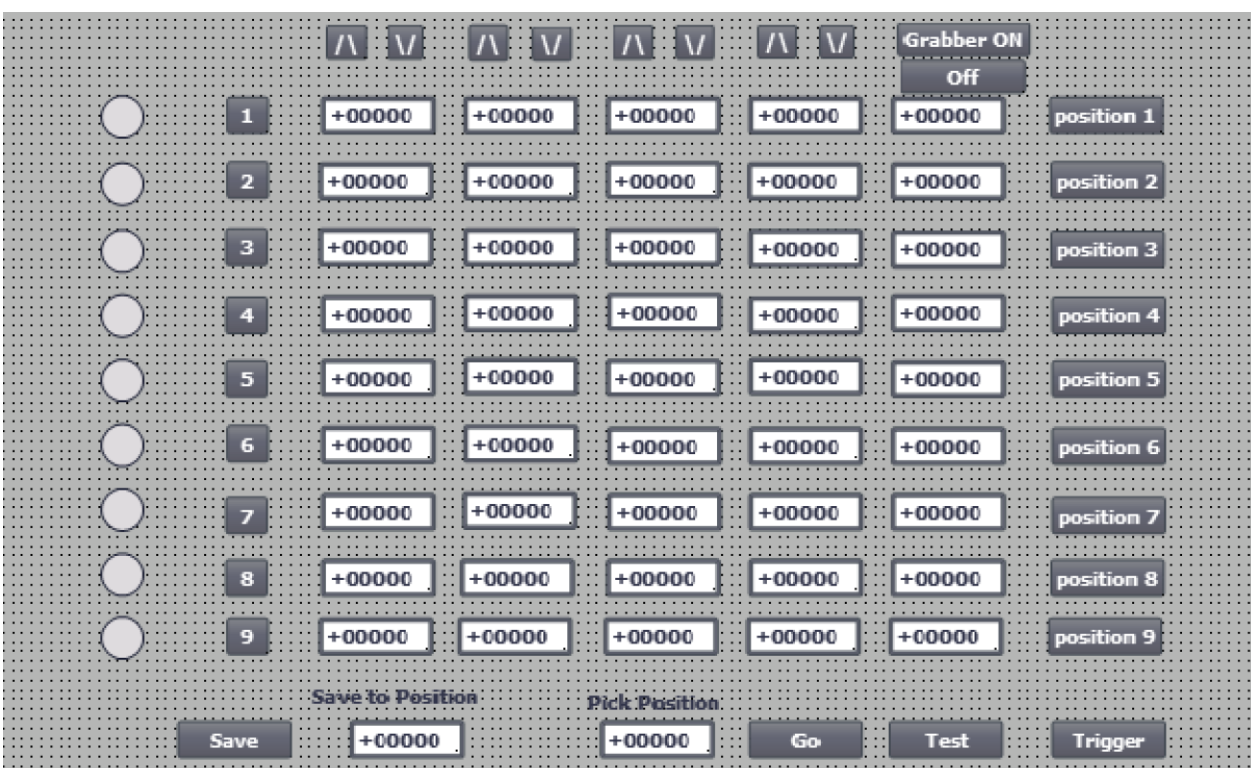

%M33.7
"x2"
MOVE
EN
ENO
IN
OUT1
"position_values".axis_4["mover1"]
"position_values".axis_4[8]
"position_values".control["pick"]
"position_values".control[0]
ADD
Auto (Int)
IN1
IN2
OUT
%MW44
"pick"
1
%MW38
"mover1"
"position_values".control["mover1"]
"position_values".control[1]
"position_values".control[2]
"position_values".control[3]
"position_values".control[4]
"position_values".control[5]
"position_values".control[6]

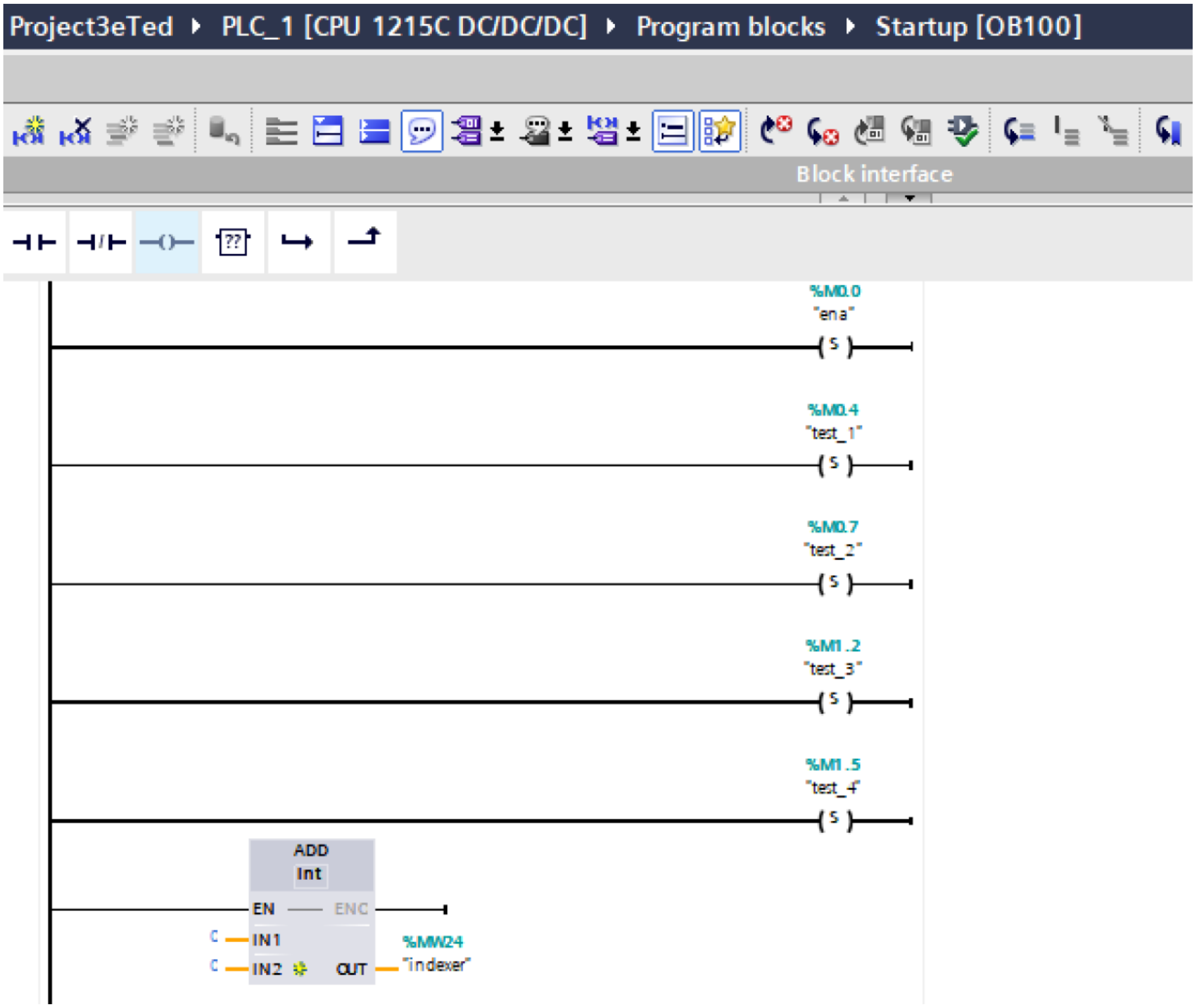



Keep actual values Snapshot Copy snapshots to start values

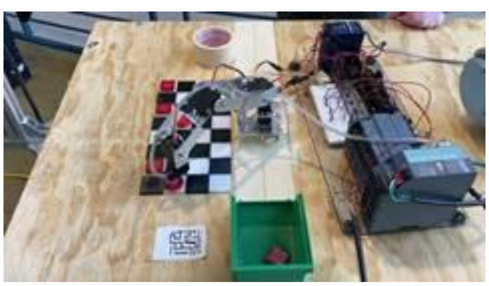
**position_values (snapshot created: 4/28/2022 6:14:26 PM)**

| | Name | Data type | Start value | Retain | Accessible f... | Writa... | Visible in ... | Setpoint | Comment |
|---|---|---|---|---|---|---|---|---|---|
| 1 | ▼ Static | | | ☐ | ☐ | ☐ | ☐ | ☐ | |
| 2 | ▶ axis_1 | Array[0..330] of... | | ☐ | ☑ | ☑ | ☑ | ☐ | |
| 3 | ▶ axis_2 | Array[0..330] of Int | | ☐ | ☑ | ☑ | ☑ | ☐ | |
| 4 | ▶ axis_3 | Array[0..330] of Int | | ☐ | ☑ | ☑ | ☑ | ☐ | |
| 5 | ▶ axis_4 | Array[0..330] of Int | | ☐ | ☑ | ☑ | ☑ | ☐ | |
| 6 | ▶ control | Array[0..330] of Int | | ☐ | ☑ | ☑ | ☑ | ☐ | |






https://youtu.be/x7R7q14b8g4



UT Senior Design Expo Spring '22 - Robotics Demonstration Cart Using Siemens PLC & Cognex Camera

The Purpose of this Senior Design Project was to integrate a Universal Robot (shown on the right side) with a Cognex Vision Camera and Siemens S7 1215C PLC (...

youtu.be