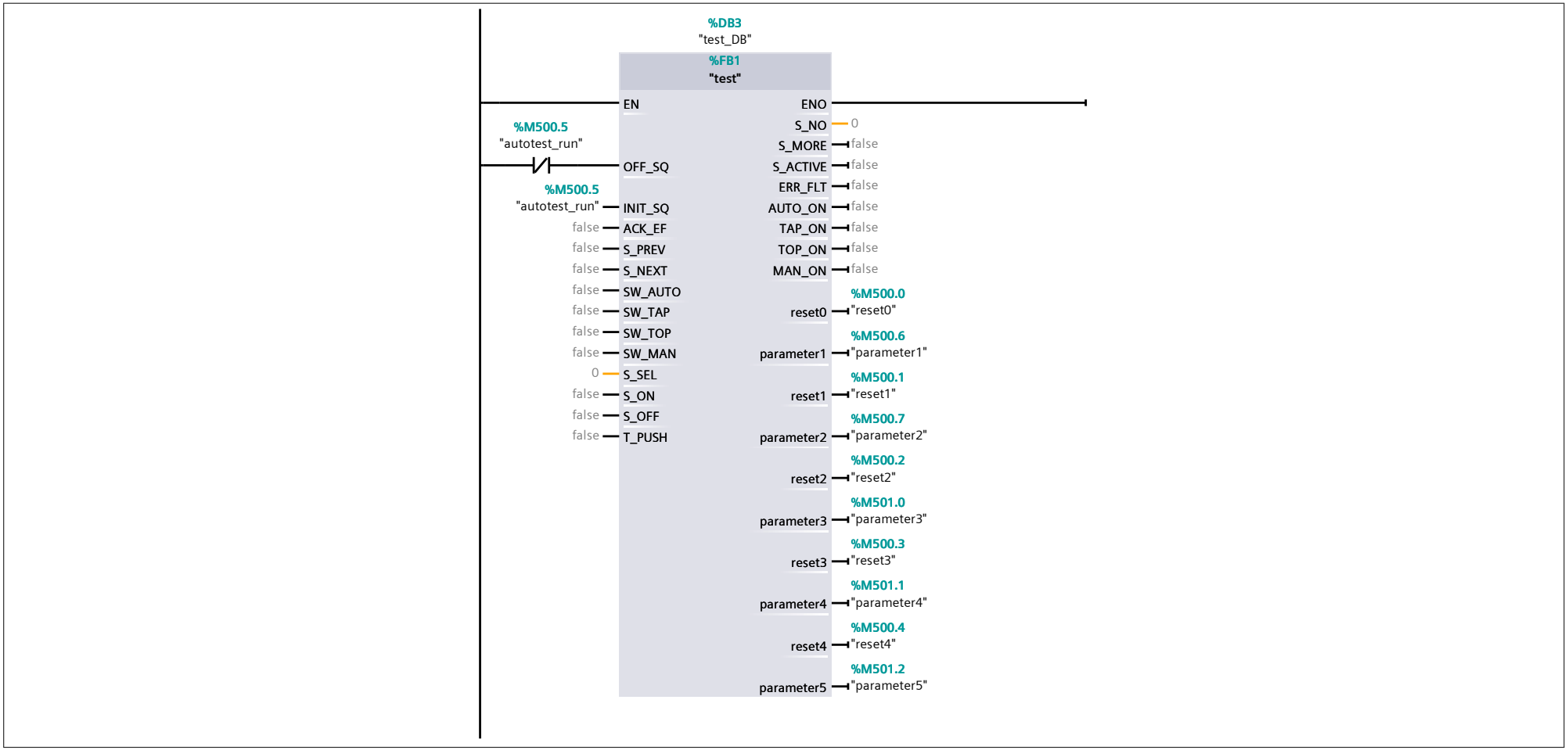
Program blocks

Main [OB1]

Main Properties							
General							
Name	Main	**Number**	1	**Type**	OB	**Language**	LAD
Numbering	Automatic						
Information							
Title	"Main Program Sweep (Cycle)"	**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
Initial_Call	Bool	
Remanence	Bool	
Temp		
Constant		

Network 1:



Network 2: test-1

rated speed of cylinder = 50mm/s [100% openning of valve]
only position loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
```

```
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 5.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 0.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
```

```
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 0.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 1.0;
0052
0053 // END_IF;
0054
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 15.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 0.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 1.0;
0065
0066 // END_IF;
0067
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 20.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 0.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 1.0;
0078
0079 // END_IF;
```

Network 3: test-2

rated speed of cylinder = 50mm/s [100% openning of valve]
position loop+velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 5.0;
```

```
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 1.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 15.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 1.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 20.0;
```

```
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 1.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 4: test-3

rated speed of cylinder = 50mm/s [100% openning of valve]
position loop+velocity loop+FF

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
```

```
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 5.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 1.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 1.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 15.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 1.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 1.0;
0065
0066 // END_IF;
```

```
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 20.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 1.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 1.0;
0078
0079 // END_IF;
```

Network 5: test-4

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
```

```
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 10.0;
```

```
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 10.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 20.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 6: test-5

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
```

```
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 1.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 1.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 1.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
```

```
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 10.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 1.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 10.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 20.0;
0076 // "vel_loop_Ti" := 1.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 7: test-5

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
```

```
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 10.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
```

```
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 10.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 10.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 10.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 10.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 20.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 8: test-7

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
```

```
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 1.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 1.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
```

```
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 2.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 10.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 5.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 10.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 5.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 9: test-9

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 1.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
```

```
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 1.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 10.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 2.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 10.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 5.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 10.0;
0074 // "pos_loop_Ti" := 0.0;
```

```
0075 // "vel_loop_kp" := 5.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 10: test-10

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 5.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 2.0;
0023 // "vel_loop_Ti" := 5.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
```

```
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 5.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 2.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 5.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 2.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 2.0;
0063 // "vel_loop_Ti" := 10.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
```

```
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 2.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 2.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 11: test-11

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 10.0;
```

```
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 1.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 2.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 1.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 1.0;
0050 // "vel_loop_Ti" := 10.0;
0051 // "vel_loop_ff_kp" := 5.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 1.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 1.0;
```

```
0063 // "vel_loop_Ti" := 10.0;
0064 // "vel_loop_ff_kp" := 10.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 1.0;
0074 // "pos_loop_Ti" := 10.0;
0075 // "vel_loop_kp" := 1.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 12: test-12

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
```

```
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 0.5;
0021 // "pos_loop_Ti" := 10.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 10.0;
0024 // "vel_loop_ff_kp" := 10.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.5;
0035 // "pos_loop_Ti" := 20.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 10.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.5;
0048 // "pos_loop_Ti" := 30.0;
0049 // "vel_loop_kp" := 1.0;
0050 // "vel_loop_Ti" := 10.0;
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
```

```
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.5;
0061 // "pos_loop_Ti" := 40.0;
0062 // "vel_loop_kp" := 1.0;
0063 // "vel_loop_Ti" := 10.0;
0064 // "vel_loop_ff_kp" := 10.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.5;
0074 // "pos_loop_Ti" := 50.0;
0075 // "vel_loop_kp" := 1.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 10.0;
0078
0079 // END_IF;
```

Network 13: test-13

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
```

```
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 0.5;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 10.0;
0024 // "vel_loop_ff_kp" := 10.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.5;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 10.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.5;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 1.0;
0050 // "vel_loop_Ti" := 10.0;
```

```
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.5;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 1.0;
0063 // "vel_loop_Ti" := 10.0;
0064 // "vel_loop_ff_kp" := 10.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.5;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 1.0;
0076 // "vel_loop_Ti" := 10.0;
0077 // "vel_loop_ff_kp" := 10.0;
0078
0079 // END_IF;
```

Network 14: test-14

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
```

```
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 0.5;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 10.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.5;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 1.0;
0037 // "vel_loop_Ti" := 10.0;
0038 // "vel_loop_ff_kp" := 10.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
```

```
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.5;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 20.0;
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.5;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 2.0;
0063 // "vel_loop_Ti" := 50.0;
0064 // "vel_loop_ff_kp" := 10.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.5;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 2.0;
0076 // "vel_loop_Ti" := 100.0;
0077 // "vel_loop_ff_kp" := 20.0;
0078
0079 // END_IF;
```

Network 15: test-15

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 0.5;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.5;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
```

```
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.5;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 10.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 20.0;
0078
```

```
0079 // END_IF;
```

Network 16: test-16

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 0.5;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
```

```
0034 // "pos_loop_kp" := 0.5;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 2.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 0.5;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 2.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
```

```
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 10.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 20.0;
0078
0079 // END_IF;
```

Network 17: test-17

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
```

```
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 20.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 10.0;
0050 // "vel_loop_Ti" := 100.0;
0051 // "vel_loop_ff_kp" := 20.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 0.5;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 100.0;
0064 // "vel_loop_ff_kp" := 20.0;
0065
0066 // END_IF;
```

```
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 0.5;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 5.0;
0076 // "vel_loop_Ti" := 50.0;
0077 // "vel_loop_ff_kp" := 20.0;
0078
0079 // END_IF;
```

Network 18: test-18

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
```

```
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 20.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 10.0;
0050 // "vel_loop_Ti" := 100.0;
0051 // "vel_loop_ff_kp" := 20.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.5;
```

```
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 100.0;
0064 // "vel_loop_ff_kp" := 20.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.5;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 5.0;
0076 // "vel_loop_Ti" := 50.0;
0077 // "vel_loop_ff_kp" := 20.0;
0078
0079 // END_IF;
```

Network 19: test-19

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
```

```
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 10.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 10.0;
0052
0053 // END_IF;
0054 // //
```

```
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 50.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 10.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 100.0;
0078
0079 // END_IF;
```

Network 20: test-20

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
```

```
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 100.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
```

```
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 100.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 100.0;
0064 // "vel_loop_ff_kp" := 100.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 5.0;
0076 // "vel_loop_Ti" := 500.0;
0077 // "vel_loop_ff_kp" := 100.0;
0078
0079 // END_IF;
```

Network 21: test-20

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
```

```
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // //
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 100.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
```

```
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 100.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 1000.0;
0064 // "vel_loop_ff_kp" := 100.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 10.0;
0076 // "vel_loop_Ti" := 5000.0;
0077 // "vel_loop_ff_kp" := 100.0;
0078
0079 // END_IF;
```

Network 22: recommandation

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 10.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
```

```
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 100.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 0.1;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 5.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 100.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 5.0;
0063 // "vel_loop_Ti" := 1000.0;
0064 // "vel_loop_ff_kp" := 100.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
```

```
0075 // "vel_loop_kp" := 10.0;
0076 // "vel_loop_Ti" := 5000.0;
0077 // "vel_loop_ff_kp" := 100.0;
0078
0079 // END_IF;
```

Network 23: 4 Paper-1

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1100.0;
0009 // "right_cylinder_pos_init" :=900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
```

```
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 0.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 30.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 0.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 1.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 50.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 0.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 1.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
```

```
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 100.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 0.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 1.0;
0078
0079 // END_IF;
```

Network 24: 4 Paper-2

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" :=1000.0;
0009 // "right_cylinder_pos_init" :=1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 0.0;
0023 // "vel_loop_Ti" := 0.0;
```

```
0024 // "vel_loop_ff_kp" := 1.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 10.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 0.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 1.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 30.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 0.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 1.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 50.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 0.0;
```

```
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 1.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 100.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 0.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 1.0;
0078
0079 // END_IF;
```

Network 25: 4 Paper-3

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" := 1000.0;
0009 // "right_cylinder_pos_init" := 1000.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
```

```
0018
0019 // "ref_velocity" := 40.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 1.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 40.0;
0047 // "pos_loop_kp" := 1.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 20.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
```

```
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 40.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 40.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 20.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 26: 4 Paper-4

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" := 1100.0;
0009 // "right_cylinder_pos_init" := 900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
```

```
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 1.0;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 1.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 0.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 0.0;
0034 // "pos_loop_kp" := 1.0;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 0.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 1.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 20.0;
0050 // "vel_loop_Ti" := 0.0;
```

```
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.1;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 10.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.1;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 20.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 27: 4 Paper-5-1

velocity loop

```
0001 // IF ("reset0" OR
0002 // "reset1" OR
0003 // "reset2" OR
0004 // "reset3" OR
0005 // "reset4")OR
```

```
0006 // NOT "autotest_run"
0007 // THEN
0008 // "left_cylinder_pos_init" := 1100.0;
0009 // "right_cylinder_pos_init" := 900.0;
0010 // "ref_velocity" := 0.0;
0011 // "model_run" := FALSE;
0012 // "controller_run" := FALSE;
0013 // END_IF;
0014 // //
0015 // IF "parameter1" THEN
0016 // "model_run" := TRUE;
0017 // "controller_run" := TRUE;
0018
0019 // "ref_velocity" := 0.0;
0020 // "pos_loop_kp" := 0.1;
0021 // "pos_loop_Ti" := 0.0;
0022 // "vel_loop_kp" := 10.0;
0023 // "vel_loop_Ti" := 0.0;
0024 // "vel_loop_ff_kp" := 100.0;
0025
0026 // END_IF;
0027
0028 // // best parameter
0029 // IF "parameter2" THEN
0030 // "model_run" := TRUE;
0031 // "controller_run" := TRUE;
0032
0033 // "ref_velocity" := 40.0;
0034 // "pos_loop_kp" := 0.1;
0035 // "pos_loop_Ti" := 0.0;
0036 // "vel_loop_kp" := 10.0;
0037 // "vel_loop_Ti" := 0.0;
0038 // "vel_loop_ff_kp" := 100.0;
0039
0040 // END_IF;
0041 // //
0042 // IF "parameter3" THEN
0043 // "model_run" := TRUE;
0044 // "controller_run" := TRUE;
```

```
0045
0046 // "ref_velocity" := 0.0;
0047 // "pos_loop_kp" := 0.0;
0048 // "pos_loop_Ti" := 0.0;
0049 // "vel_loop_kp" := 0.0;
0050 // "vel_loop_Ti" := 0.0;
0051 // "vel_loop_ff_kp" := 0.0;
0052
0053 // END_IF;
0054 // //
0055 // IF "parameter4" THEN
0056 // "model_run" := TRUE;
0057 // "controller_run" := TRUE;
0058
0059 // "ref_velocity" := 0.0;
0060 // "pos_loop_kp" := 0.0;
0061 // "pos_loop_Ti" := 0.0;
0062 // "vel_loop_kp" := 0.0;
0063 // "vel_loop_Ti" := 0.0;
0064 // "vel_loop_ff_kp" := 0.0;
0065
0066 // END_IF;
0067 // //
0068 // IF "parameter5" THEN
0069 // "model_run" := TRUE;
0070 // "controller_run" := TRUE;
0071
0072 // "ref_velocity" := 0.0;
0073 // "pos_loop_kp" := 0.0;
0074 // "pos_loop_Ti" := 0.0;
0075 // "vel_loop_kp" := 0.0;
0076 // "vel_loop_Ti" := 0.0;
0077 // "vel_loop_ff_kp" := 0.0;
0078
0079 // END_IF;
```

Network 28: 4 Paper-5-2

velocity loop

```
0001 IF ("reset0" OR
0002     "reset1" OR
0003     "reset2" OR
0004     "reset3" OR
0005     "reset4") OR
0006     NOT "autotest_run"
0007 THEN
0008     "left_cylinder_pos_init" := 1100.0;
0009     "right_cylinder_pos_init" := 900.0;
0010     "ref_velocity" := 0.0;
0011     "model_run" := FALSE;
0012     "controller_run" := FALSE;
0013 END_IF;
0014 //
0015 IF "parameter1" THEN
0016     "model_run" := TRUE;
0017     "controller_run" := TRUE;
0018
0019     "ref_velocity" := 40.0;
0020     "pos_loop_kp" := 0.1;
0021     "pos_loop_Ti" := 0.0;
0022     "vel_loop_kp" := 10.0;
0023     "vel_loop_Ti" := 10.0;
0024     "vel_loop_ff_kp" := 100.0;
0025
0026 END_IF;
0027
0028 // best parameter
0029 IF "parameter2" THEN
0030     "model_run" := TRUE;
0031     "controller_run" := TRUE;
0032
0033     "ref_velocity" := 40.0;
0034     "pos_loop_kp" := 0.1;
0035     "pos_loop_Ti" := 0.0;
0036     "vel_loop_kp" := 10.0;
0037     "vel_loop_Ti" := 0.0;
0038     "vel_loop_ff_kp" := 100.0;
0039
```

```
0040 END_IF;
0041 //
0042 IF "parameter3" THEN
0043     "model_run" := TRUE;
0044     "controller_run" := TRUE;
0045
0046     "ref_velocity" := 0.0;
0047     "pos_loop_kp" := 0.0;
0048     "pos_loop_Ti" := 0.0;
0049     "vel_loop_kp" := 0.0;
0050     "vel_loop_Ti" := 0.0;
0051     "vel_loop_ff_kp" := 0.0;
0052
0053 END_IF;
0054 //
0055 IF "parameter4" THEN
0056     "model_run" := TRUE;
0057     "controller_run" := TRUE;
0058
0059     "ref_velocity" := 0.0;
0060     "pos_loop_kp" := 0.0;
0061     "pos_loop_Ti" := 0.0;
0062     "vel_loop_kp" := 0.0;
0063     "vel_loop_Ti" := 0.0;
0064     "vel_loop_ff_kp" := 0.0;
0065
0066 END_IF;
0067 //
0068 IF "parameter5" THEN
0069     "model_run" := TRUE;
0070     "controller_run" := TRUE;
0071
0072     "ref_velocity" := 0.0;
0073     "pos_loop_kp" := 0.0;
0074     "pos_loop_Ti" := 0.0;
0075     "vel_loop_kp" := 0.0;
0076     "vel_loop_Ti" := 0.0;
0077     "vel_loop_ff_kp" := 0.0;
0078
```

```
0079 END_IF;
```

Program blocks

test [FB1]

test Properties							
General							
Name	test	**Number**	1	**Type**	FB	**Language**	GRAPH
Numbering	Automatic	**Network language**	LAD	**Block version**	V6.0		
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
OFF_SQ	Bool	false	Non-retain
INIT_SQ	Bool	false	Non-retain
ACK_EF	Bool	false	Non-retain
S_PREV	Bool	false	Non-retain
S_NEXT	Bool	false	Non-retain
SW_AUTO	Bool	false	Non-retain
SW_TAP	Bool	false	Non-retain
SW_TOP	Bool	false	Non-retain
SW_MAN	Bool	false	Non-retain
S_SEL	Int	0	Non-retain
S_ON	Bool	false	Non-retain
S_OFF	Bool	false	Non-retain
T_PUSH	Bool	false	Non-retain
▼ Output			
S_NO	Int	0	Non-retain
S_MORE	Bool	false	Non-retain
S_ACTIVE	Bool	false	Non-retain
ERR_FLT	Bool	false	Non-retain
AUTO_ON	Bool	false	Non-retain
TAP_ON	Bool	false	Non-retain

Name	Data type	Default value	Retain
TOP_ON	Bool	false	Non-retain
MAN_ON	Bool	false	Non-retain
reset0	Bool	false	Non-retain
parameter1	Bool	false	Non-retain
reset1	Bool	false	Non-retain
parameter2	Bool	false	Non-retain
reset2	Bool	false	Non-retain
parameter3	Bool	false	Non-retain
reset3	Bool	false	Non-retain
parameter4	Bool	false	Non-retain
reset4	Bool	false	Non-retain
parameter5	Bool	false	Non-retain
InOut			
▼ Static			
RT_DATA	G7_RTDataPlus_V6		Non-retain
Trans1	G7_TransitionPlus_V6		Non-retain
Trans2	G7_TransitionPlus_V6		Non-retain
Trans3	G7_TransitionPlus_V6		Non-retain
Trans4	G7_TransitionPlus_V6		Non-retain
Trans5	G7_TransitionPlus_V6		Non-retain
Trans6	G7_TransitionPlus_V6		Non-retain
Trans7	G7_TransitionPlus_V6		Non-retain
Trans8	G7_TransitionPlus_V6		Non-retain
Trans9	G7_TransitionPlus_V6		Non-retain
Trans10	G7_TransitionPlus_V6		Non-retain
Step1	G7_StepPlus_V6		Non-retain
Step2	G7_StepPlus_V6		Non-retain
Step3	G7_StepPlus_V6		Non-retain
Step4	G7_StepPlus_V6		Non-retain
Step5	G7_StepPlus_V6		Non-retain
Step6	G7_StepPlus_V6		Non-retain
Step7	G7_StepPlus_V6		Non-retain
Step8	G7_StepPlus_V6		Non-retain
Step9	G7_StepPlus_V6		Non-retain

Name	Data type	Default value	Retain
Step10	G7_StepPlus_V6		Non-retain
Temp			
▼ Constant			
MIN_INTERVAL	Time	T#10ms	
MAX_INTERVAL	Time	T#40s	
LAST_INTERVAL	Time	T#60s	

Alarms

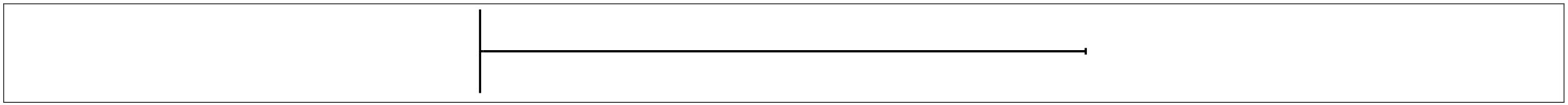
Enable alarms	True

Category	Category enabler	Display class
错误		0
警告		0
信息		0
类别 4		0
类别 5		0
类别 6		0
类别 7		0
类别 8		0

Category for interlocks	错误	**Subcategory 1 for interlocks**		**Subcategory 2 for interlocks**	
Category for supervisions	错误	**Subcategory 1 for supervisions**		**Subcategory 2 for supervisions**	
Category for GRAPH warnings	警告	**Subcategory 1 for GRAPH warnings**		**Subcategory 2 for GRAPH warnings**	

Permanent pre-instructions

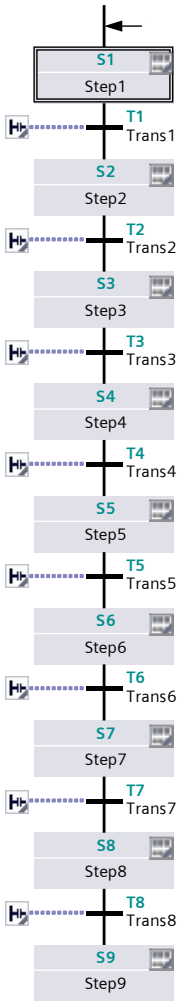
1:



Sequences (1)

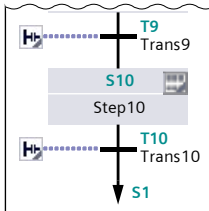
1:

1: (1.1 / 2.1)


S1
Step1
T1
Trans1
S2
Step2
T2
Trans2
S3
Step3
T3
Trans3
S4
Step4
T4
Trans4
S5
Step5
T5
Trans5
S6
Step6
T6
Trans6
S7
Step7
T7
Trans7
S8
Step8
T8
Trans8
S9
Step9

1: (2.1 / 2.1)

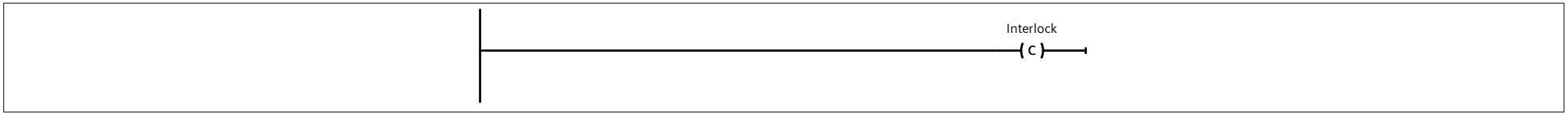
1.1 (Page2 - 5)



S1 - [Initial step]:Step1

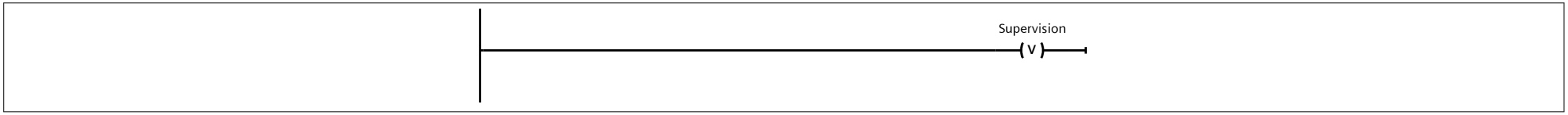
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

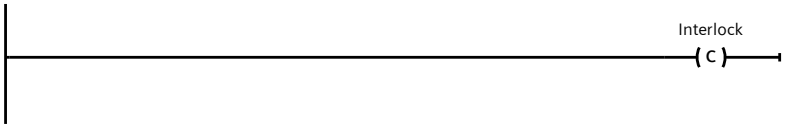
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#reset0

T1:Trans1

S2:Step2

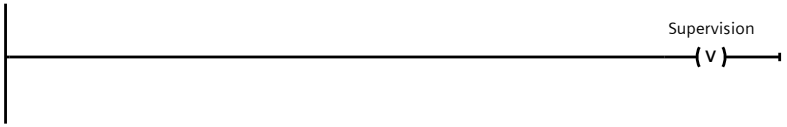
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

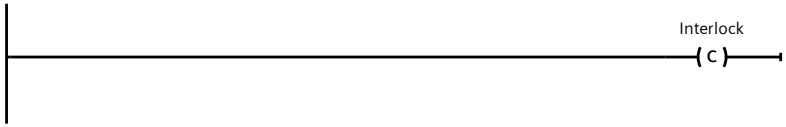
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#parameter1

T2:Trans2

S3:Step3

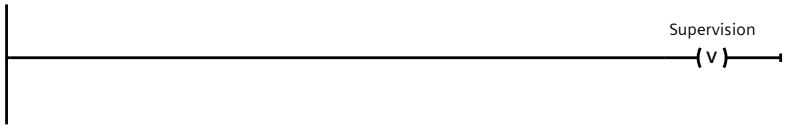
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

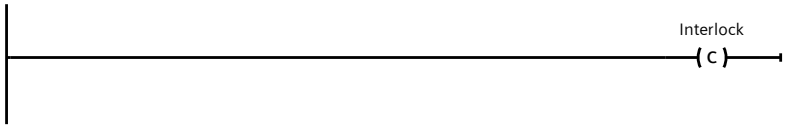
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#reset1

T3:Trans3

S4:Step4

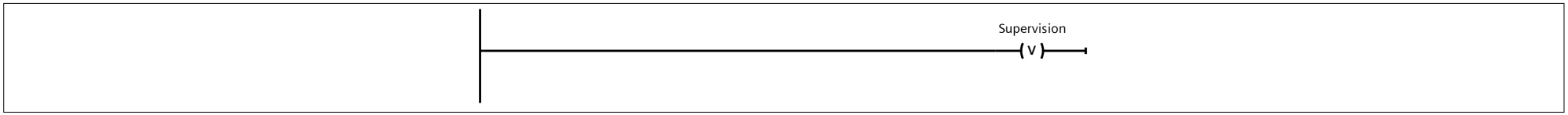
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

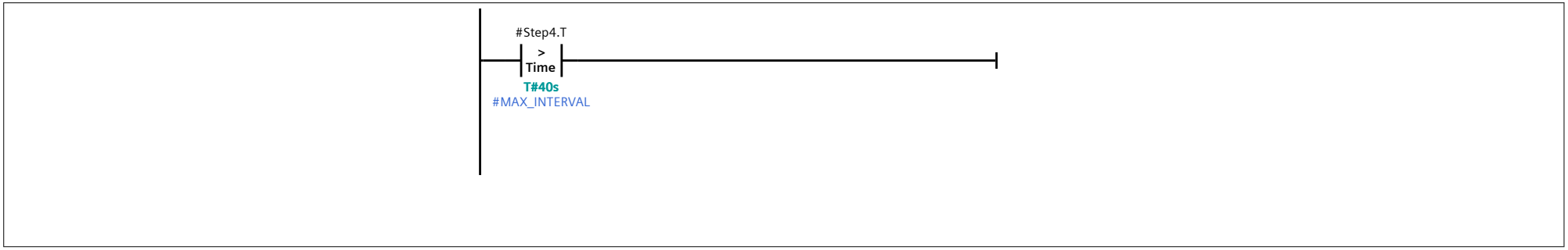
Supervision alarm	
Alarm text	



Actions:

Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#parameter2

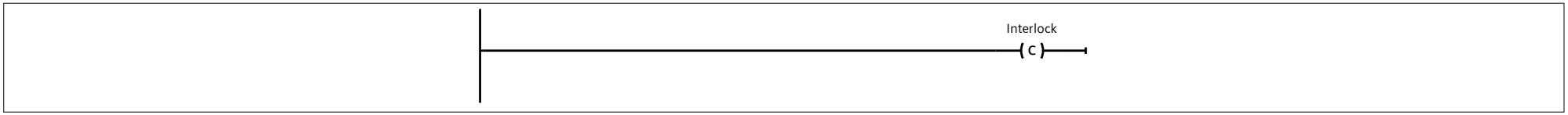
T4:Trans4



S5:Step5

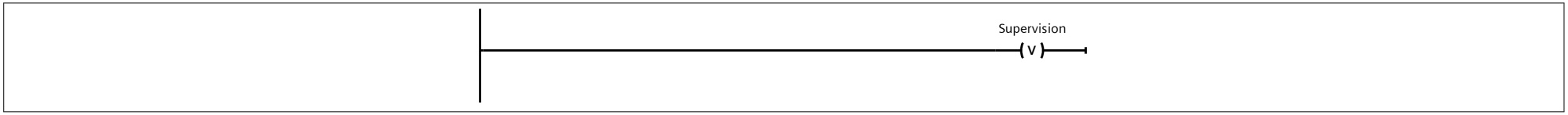
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

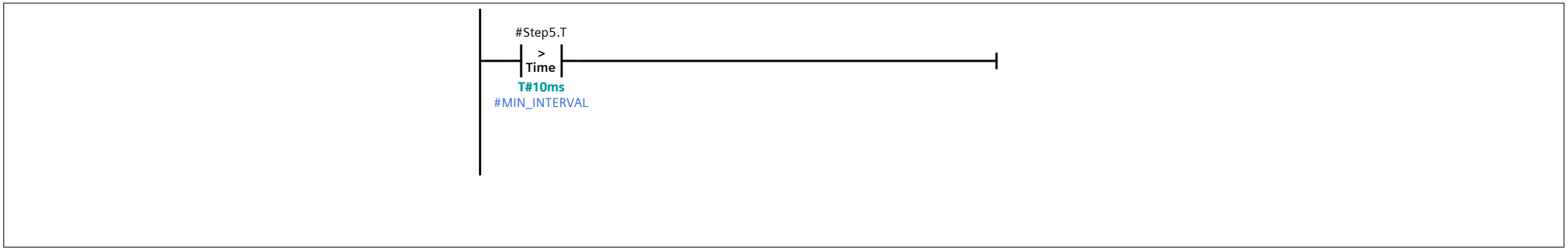
Supervision alarm	
Alarm text	



Actions:

Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#reset2

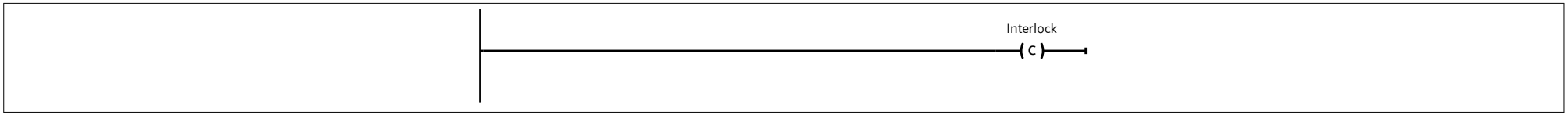
T5:Trans5



S6:Step6

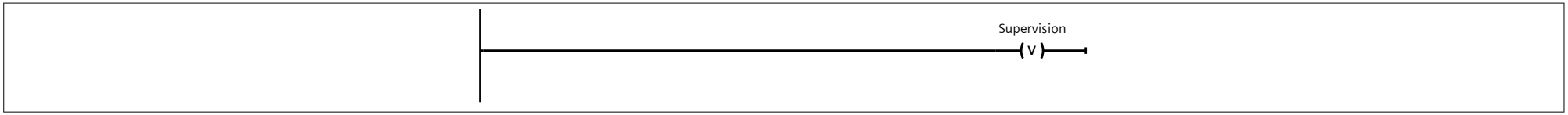
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

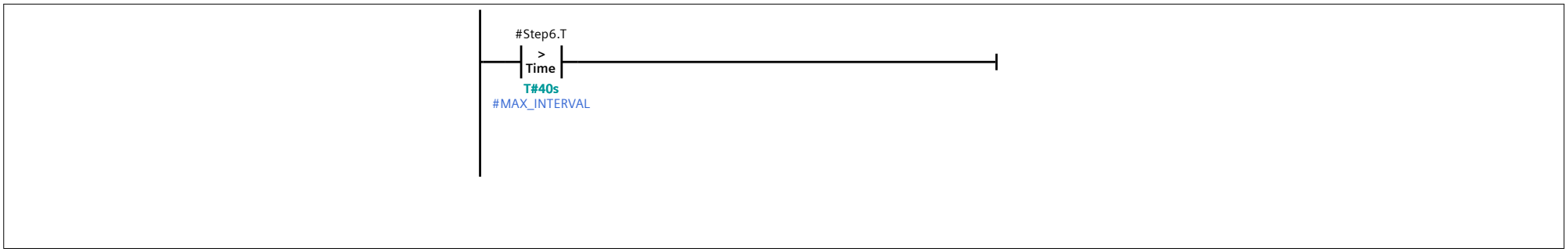
Supervision alarm	
Alarm text	



Actions:

Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#parameter3

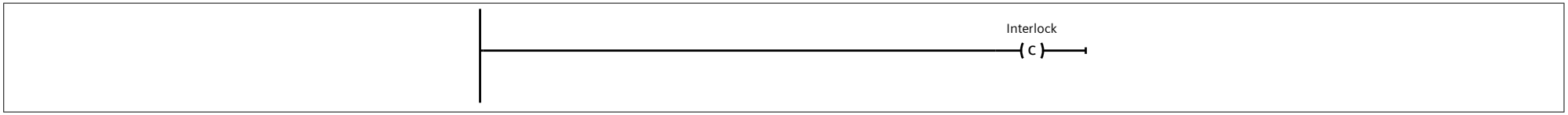
T6:Trans6



S7:Step7

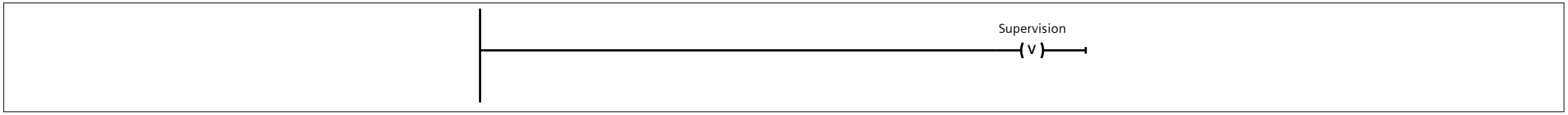
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

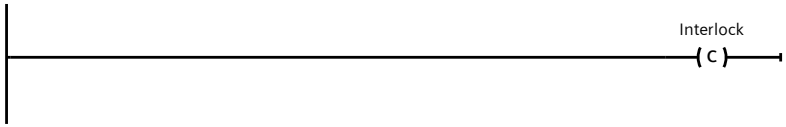
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#reset3

T7:Trans7

S8:Step8

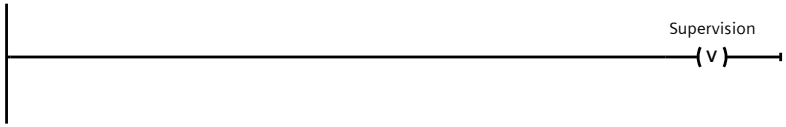
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

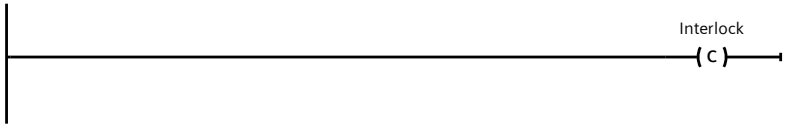
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#parameter4

T8:Trans8

S9:Step9

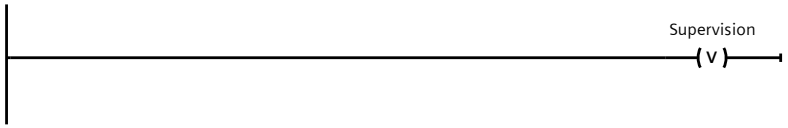
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

Supervision alarm	
Alarm text	



Actions:

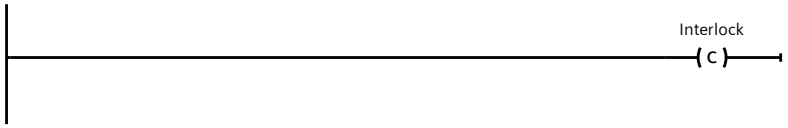
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#reset4

T9:Trans9

S10:Step10

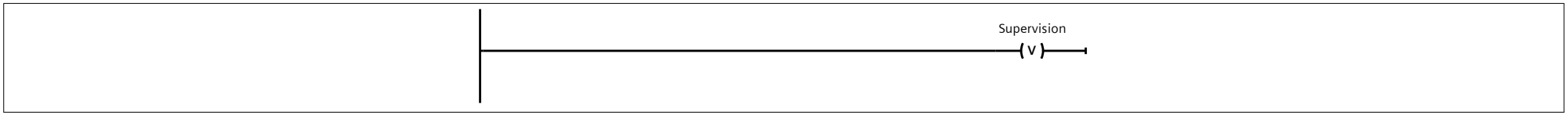
Interlock -(c)-:

Interlock alarm	
Alarm text	



Supervision -(v)-:

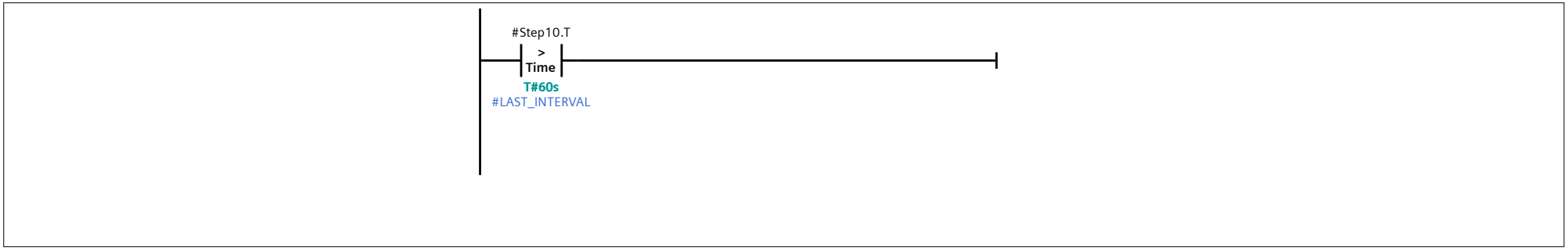
Supervision alarm	
Alarm text	



Actions:

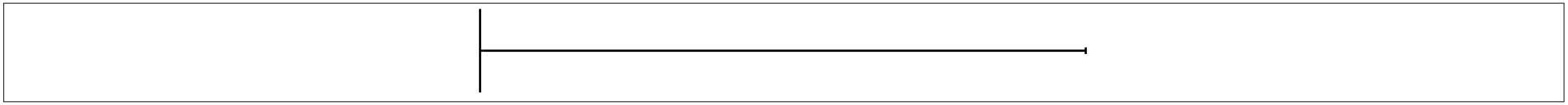
Actions:			
Interlock	**Event**	**Qualifier**	**Action**
		N	#parameter5

T10:Trans10



Permanent post-instructions

1:



Program blocks

test_DB [DB3]

test_DB Properties							
General							
Name	test_DB	**Number**	3	**Type**	DB	**Language**	DB
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Input			
OFF_SQ	Bool	false	False
INIT_SQ	Bool	false	False
ACK_EF	Bool	false	False
S_PREV	Bool	false	False
S_NEXT	Bool	false	False
SW_AUTO	Bool	false	False
SW_TAP	Bool	false	False
SW_TOP	Bool	false	False
SW_MAN	Bool	false	False
S_SEL	Int	0	False
S_ON	Bool	false	False
S_OFF	Bool	false	False
T_PUSH	Bool	false	False
▼ Output			
S_NO	Int	0	False
S_MORE	Bool	false	False
S_ACTIVE	Bool	false	False
ERR_FLT	Bool	false	False
AUTO_ON	Bool	false	False
TAP_ON	Bool	false	False

Name	Data type	Start value	Retain
TOP_ON	Bool	false	False
MAN_ON	Bool	false	False
reset0	Bool	false	False
parameter1	Bool	false	False
reset1	Bool	false	False
parameter2	Bool	false	False
reset2	Bool	false	False
parameter3	Bool	false	False
reset3	Bool	false	False
parameter4	Bool	false	False
reset4	Bool	false	False
parameter5	Bool	false	False
InOut			
▼ Static			
RT_DATA	G7_RTDataPlus_V6		False
Trans1	G7_TransitionPlus_V6		False
Trans2	G7_TransitionPlus_V6		False
Trans3	G7_TransitionPlus_V6		False
Trans4	G7_TransitionPlus_V6		False
Trans5	G7_TransitionPlus_V6		False
Trans6	G7_TransitionPlus_V6		False
Trans7	G7_TransitionPlus_V6		False
Trans8	G7_TransitionPlus_V6		False
Trans9	G7_TransitionPlus_V6		False
Trans10	G7_TransitionPlus_V6		False
Step1	G7_StepPlus_V6		False
Step2	G7_StepPlus_V6		False
Step3	G7_StepPlus_V6		False
Step4	G7_StepPlus_V6		False
Step5	G7_StepPlus_V6		False
Step6	G7_StepPlus_V6		False
Step7	G7_StepPlus_V6		False
Step8	G7_StepPlus_V6		False
Step9	G7_StepPlus_V6		False

Name	Data type	Start value	Retain
Step10	G7_StepPlus_V6		False

Program blocks

test_snap7 [DB1]

test_snap7 Properties							
General							
Name	test_snap7	**Number**	1	**Type**	DB	**Language**	DB
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Static			
Static_1	Bool	false	False
world	Bool	false	False
my	Byte	16#0	False
name	Word	16#0	False
is	DWord	16#0	False
g	Int	0	False
q	Real	0.0	False
q1	Real	0.0	False
q2	Real	0.0	False
q3	Real	0.0	False
q4	Real	0.0	False
q5	Real	0.0	False

Program blocks

test_snap7_abs [DB2]

test_snap7_abs Properties							
General							
Name	test_snap7_abs	**Number**	2	**Type**	DB	**Language**	DB
Numbering	Manual						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Static			
first	Bool	false	False
name	Bool	false	False

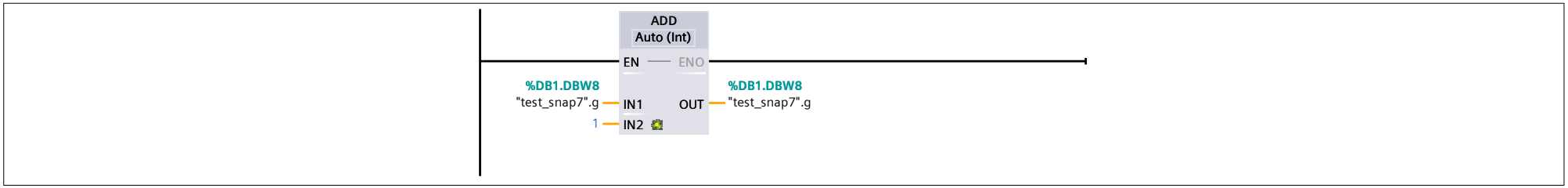
Program blocks

Cyclic interrupt [OB32]

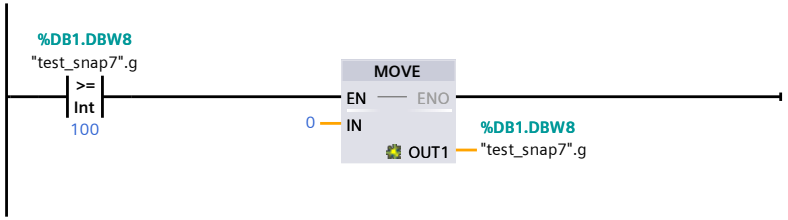
Cyclic interrupt Properties							
General							
Name	Cyclic interrupt	**Number**	32	**Type**	OB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
Input		
Initial_Call	Bool	
Event_Count	Int	
Temp		
Constant		

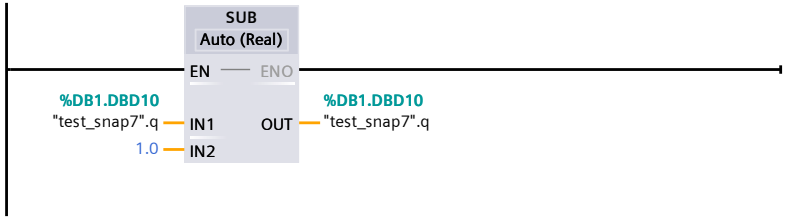
Network 1:



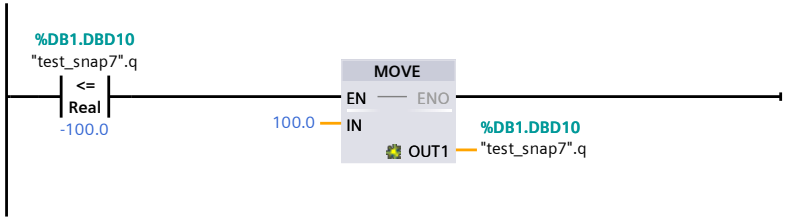
Network 2:



Network 3:



Network 4:



Program blocks / lib

DeadBand_Real [FC30005]

DeadBand_Real Properties							
General							
Name	DeadBand_Real	**Number**	30005	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_R_X	Real	
i_R_DBWidth	Real	
▼ Output		
o_R_Y	Real	
InOut		
▼ Temp		
i_R_AbsX	Real	
i_R_AbsDBWidth	Real	
Constant		
▼ Return		
DeadBand_Real	Void	

```
0001 #i_R_AbsX := ABS(#i_R_X);
0002 #i_R_AbsDBWidth := ABS(#i_R_DBWidth);
0003 IF #i_R_AbsX > #i_R_AbsDBWidth AND #i_R_X >= 0 THEN
0004   #o_R_Y := #i_R_X - #i_R_AbsDBWidth;
0005 ELSIF #i_R_AbsX > #i_R_AbsDBWidth AND #i_R_X < 0 THEN
0006   #o_R_Y := #i_R_X + #i_R_AbsDBWidth;
0007 ELSE
0008   #o_R_Y := REAL#0.0;
0009 END_IF;
```

```
0010
```

Program blocks / lib

DELAY [FB2]

DELAY Properties							
General							
Name	DELAY	**Number**	2	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
RUN	Bool	false	Non-retain
XIN	Real	0.0	Non-retain
N	Int	0	Non-retain
▼ Output			
XOUT	Real	0.0	Non-retain
InOut			
▼ Static			
X	Array[0..127] of Real		Non-retain
I	Int	0	Non-retain
ID	Int	0	Non-retain
Temp			
Constant			

Network 1:

```
0001 IF #RUN THEN
0002     FOR #ID := 127 TO 1 BY -1 DO
0003         #X[#ID] := #X[#ID - 1];
0004     END_FOR;
0005     #X[0] := #XIN;
```

```
0006      #XOUT := #X[#N];
0007  ELSE
0008      #XOUT := #XIN;
0009      FOR #I := 0 TO #N DO
0010          #X[#I] := #XIN;
0011      END_FOR;
0012  END_IF;
```

Program blocks / lib

DERIVATIVE [FB6]

DERIVATIVE Properties							
General							
Name	DERIVATIVE	**Number**	6	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
RUN	Bool	false	Non-retain
XIN	Real	0.0	Non-retain
CYCLE	Real	0.0	Non-retain
▼ Output			
XOUT	Real	0.0	Non-retain
InOut			
▼ Static			
X1	Real	0.0	Non-retain
X2	Real	0.0	Non-retain
X3	Real	0.0	Non-retain
Temp			
Constant			

Network 1: Differentiated output

```
0001 IF #RUN THEN
0002     #XOUT := (3.0 * (#XIN - #X3) + #X1 - #X2) / (10.0 * #CYCLE);
0003     #X3 := #X2;
0004     #X2 := #X1;
0005     #X1 := #XIN;
```

```
0006 ELSE
0007     #XOUT := 0.0;
0008     #X1 := #XIN;
0009     #X2 := #XIN;
0010     #X3 := #XIN;
0011 END_IF;
```

Program blocks / lib

INTEGRAL [FB5]

INTEGRAL Properties							
General							
Name	INTEGRAL	**Number**	5	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
RUN	Bool	false	Non-retain
R1	Bool	false	Non-retain
XIN	Real	0.0	Non-retain
X0	Real	0.0	Non-retain
CYCLE	Real	0.0	Non-retain
▼ Output			
Q	Bool	false	Non-retain
XOUT	Real	0.0	Non-retain
InOut			
Static			
Temp			
Constant			

Network 1: NOT R1

```
0001 #Q := NOT #R1;
0002 IF #R1 THEN #XOUT := #X0;
0003 ELSIF #RUN THEN
0004     #XOUT := #XOUT + #XIN *#CYCLE;
```

```
0005 END_IF;
```

Program blocks / lib

LAG1 [FB3]

LAG1 Properties							
General							
Name	LAG1	**Number**	3	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
RUN	Bool	false	Non-retain
XIN	Real	0.0	Non-retain
TAU	Real	0.0	Non-retain
CYCLE	Real	0.0	Non-retain
▼ Output			
XOUT	Real	0.0	Non-retain
InOut			
▼ Static			
K	Real	0.0	Non-retain
Temp			
Constant			

Network 1:

```
0001 IF #RUN THEN
0002     #XOUT := #XOUT + #K * (#XIN - #XOUT);
0003 ELSE
0004     #XOUT := #XIN;
0005     #K := #CYCLE / (#CYCLE + #TAU);
```

```
0006 END_IF;
```

Program blocks / lib

RateLimiter [FB9]

RateLimiter Properties							
General							
Name	RateLimiter	**Number**	9	**Type**	FB	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
i_R_X	Real	0.0	Non-retain
i_R_RateLimit	Real	0.0	Non-retain
▼ Output			
o_R_Y	Real	0.0	Non-retain
InOut			
▼ Static			
s_R_Y	Real	0.0	Non-retain
s_R_B4Y	Real	0.0	Non-retain
▼ Temp			
t_R_Diff	Real		
t_R_AbsDiff	Real		
t_R_AbsRateLimit	Real		
t_R_MinusRateLimit	Real		
Constant			

```
0001
0002 // Difference = IN - Yn-1
0003 #t_R_Diff := (#i_R_X - #s_R_B4Y);
0004 #t_R_AbsDiff := ABS(#t_R_Diff);
0005 #t_R_AbsRateLimit := ABS(#i_R_RateLimit);
0006 #t_R_MinusRateLimit := (-1.0) * #i_R_RateLimit;
```

```
0007 // When +|D| ≥ Difference ≥ -|D|: Y = IN
0008 // When +|D| < Difference: Y = Yn-1 + |D|
0009 // When -|D| > Difference: Y = Yn-1 - |D|
0010 IF
0011   #t_R_AbsDiff <= #t_R_AbsRateLimit
0012 THEN
0013   #s_R_Y := #i_R_X;
0014 ELSIF
0015   #t_R_AbsRateLimit < #t_R_Diff
0016 THEN
0017   #s_R_Y := #s_R_B4Y + #t_R_AbsRateLimit;
0018 ELSIF
0019   #t_R_Diff < #t_R_MinusRateLimit
0020 THEN
0021   #s_R_Y := #s_R_B4Y - #t_R_AbsRateLimit;
0022 ELSE
0023   #s_R_Y := #i_R_X;
0024 END_IF;
0025 #s_R_B4Y := #s_R_Y;
0026
0027 #o_R_Y := #s_R_Y;
```

Program blocks / lib

PID [FB10]

PID Properties							
General							
Name	PID	**Number**	10	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title	PID Controller	**Author**		**Comment**	2021/10/14 Created by GQ	**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
AUTO	Bool	false	Non-retain
DBW	Real	0.0	Non-retain
SP	Real	0.0	Non-retain
PV	Real	0.0	Non-retain
X0	Real	0.0	Non-retain
KP	Real	0.0	Non-retain
TR[s]	Real	0.0	Non-retain
TD[s]	Real	0.0	Non-retain
CYCLE[s]	Real	0.0	Non-retain
▼ Output			
XOUT	Real	0.0	Non-retain
STATUSINFO	Word	16#0	Non-retain
InOut			
▼ Static			
ERROR	Real	0.0	Non-retain
EDBW	Real	0.0	Non-retain
PIDOUT	Real	0.0	Non-retain
ITERM	"INTEGRAL"		
DTERM	"DERIVATIVE"		
Temp			

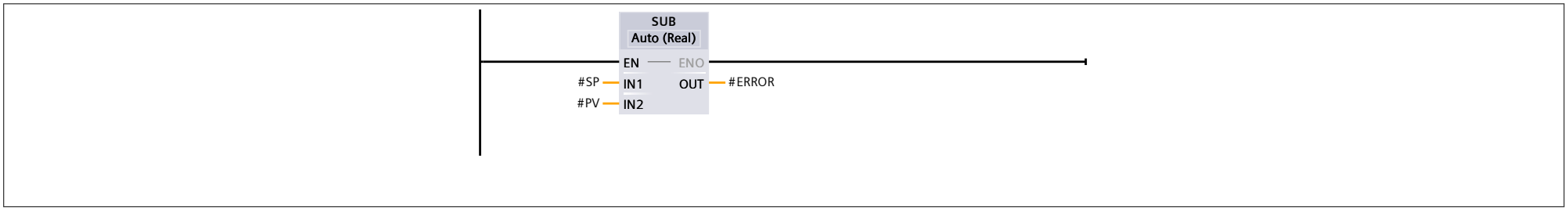
Name	Data type	Default value	Retain
Constant			
NO_ERROR	Word	16#0000	
I_TIME	Word	16#7000	
OUT_ARRAY_INDEX	Word	16#8000	
LIMIT_PARA_FAULT	Word	16#8200	
RATEVALUE_ZERO	Word	16#8400	

Network 1: License

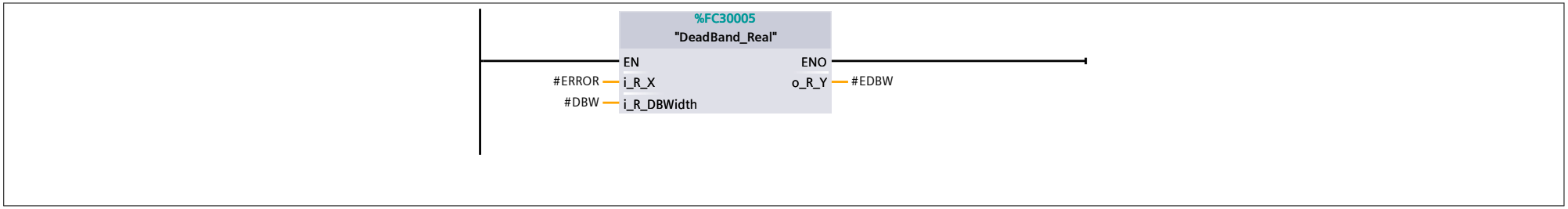
```
0001 //=============================================================================
0002 // (c)Copyright 2021
0003 //-----------------------------------------------------------------------------
0004 // Library:
0005 // Tested with: PLCSim Advanced
0006 // Engineering: TIA Portal V15 Update 1
0007 // Restrictions:
0008 // Requirements: PLC (S7-1200 / S7-1500)
0009 // Functionality: PID Controller for check result of smith predictor
0010 //-----------------------------------------------------------------------------
0011 // Change log table:
0012 // Version Date In charge / Changes applied
0013 // 00.00.01 10.14.2021 002706 created
0014 //=============================================================================
0015
0016
0017 IF #"TR[s]" < 0.0 THEN
0018     #STATUSINFO := #I_TIME;
0019     RETURN;
0020 END_IF;
0021
0022 // IF #MX < #MN THEN
0023 // #STATUSINFO := #LIMIT_PARA_FAULT;
0024 // RETURN
0025 // ;
0026 // END_IF;
0027
```

```
0028 // IF #RATEVALUE = 0.0 OR #RATEVALUE > (#MX - #MN)THEN
0029 // #STATUSINFO := #RATEVALUE_ZERO;
0030 // RETURN
0031 // ;
0032 // END_IF;
0033
0034 #STATUSINFO := #NO_ERROR;
```

Network 2: ERROR



Network 3: Deadband



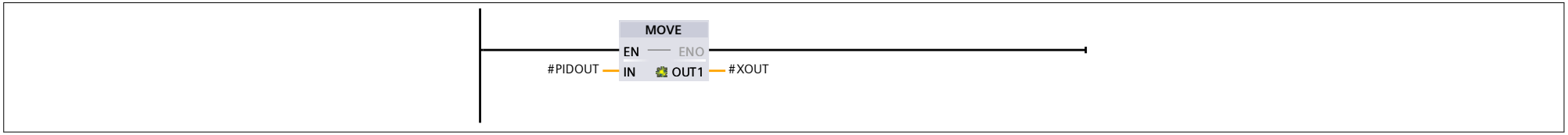
Network 4: PID

https://blog.opticontrols.com/archives/383
The Cohen-Coon tuning rules were designed for controllers with the noninteractive controller algorithm.
Different PID algorithm equation will cause different result.

```
0001 // PID
```

```
0002 (*** Adjust ITERM so that XOUT := X0 when AUTO = 0 ***)
0003 IF #KP = 0.0 THEN
0004     #PIDOUT := 0.0;
0005 ELSE
0006     #ITERM(RUN := #AUTO,
0007            R1 := NOT #AUTO,
0008            XIN := #EDBW,
0009            X0 := #"TR[s]" * (#X0 - #EDBW),
0010            CYCLE := #"CYCLE[s]");
0011     #DTERM(RUN := #AUTO,
0012            XIN := #EDBW,
0013            CYCLE := #"CYCLE[s]");
0014     IF NOT #AUTO THEN
0015         #PIDOUT := #X0;
0016     ELSIF #"TR[s]" = 0.0 THEN
0017         #PIDOUT := #KP * (#EDBW + #DTERM.XOUT * #"TD[s]");
0018     ELSE
0019         #PIDOUT := #KP * (#EDBW + #ITERM.XOUT / #"TR[s]" + #DTERM.XOUT * #"TD[s]");
0020     END_IF;
0021 END_IF;
0022
```

Network 5: output



Program blocks / lib

LGF_RandomInt [FC10013]

LGF_RandomInt Properties							
General							
Name	LGF_RandomInt	**Number**	10013	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version		**User-defined ID**					

Name	Data type	Default value
▼ Input		
minValue	Int	
maxValue	Int	
▼ Output		
error	Bool	
statusID	UInt	
status	Word	
InOut		
▼ Temp		
tempTime	DTL	
tempTimeStatus	Int	
tempRandomValue	Int	
tempNormReal	Real	
▼ Constant		
NO_ERROR	Word	16#0000
NO_CURRENT_JOBS	Word	16#7000
MAX_LESS_MIN	Word	16#8200
ERROR_IN_THIS_BLOCK	UInt	1
ERROR_RD_SYS_T	UInt	2
MIN_INT	DInt	-32768
MAX_INT	DInt	32767

Name	Data type	Default value
Return		
LGF_RandomInt	Int	

```
0001 //=============================================================================
0002 // Siemens AG
0003 // (c)Copyright 2017
0004 //-----------------------------------------------------------------------------
0005 // Library: LGF (Library General Functions)
0006 // Tested with: CPU1212C DC/DC/DC FW:V4.2
0007 // Engineering: TIA Portal V13 SP1 Upd 4
0008 // Restrictions: -
0009 // Requirements: PLC (S7-1200 / S7-1500)
0010 // Functionality: This function generates random numbers in defined limits
0011 // (Datatype Int)
0012 //-----------------------------------------------------------------------------
0013 // Change log table:
0014 // Version Date In charge / Changes applied
0015 // 01.00.00 19.08.2015 Siemens Industry Online Support
0016 // First released version
0017 // 01.00.01 02.01.2017 Siemens Industry Online Support
0018 // Upgrade: TIA Portal V14 Update 1
0019 // 01.00.01 10.18.2021 Scale_X 替换成公式 OUT = [VALUE * (MAX - MIN)] + MIN
0020 //-----------------------------------------------------------------------------
0021 //Status Codes:
0022 // 16#7000: No current jobs
0023 // 16#0000: Job finished; Note: There is no "Busy"-Status because the block
0024 // is finished within a single cycle
0025 // 16#8200: maxValue is samller then minValue
0026 //=============================================================================
0027
0028 //Set "No current job" status
0029 #error := false;
0030 #statusID := #ERROR_IN_THIS_BLOCK;
0031 #status := #NO_CURRENT_JOBS;
0032
0033 //Check if the maximal Value is less than the minimal value
0034 IF (#minValue > #maxValue) THEN
```

```
0035    #error := true;
0036    #statusID := #ERROR_IN_THIS_BLOCK;
0037    #status := #MAX_LESS_MIN;
0038    #LGF_RandomInt := 0;
0039    RETURN;
0040  END_IF;
0041
0042  //Read system time
0043  #tempTimeStatus := RD_SYS_T(#tempTime);
0044
0045  IF (#tempTimeStatus <> 0) THEN
0046    #error := true;
0047    #statusID := #ERROR_RD_SYS_T;
0048    #status := INT_TO_WORD(#tempTimeStatus);
0049    #LGF_RandomInt := 0;
0050    RETURN;
0051  END_IF;
0052
0053  //Callculate a random-start-value depending on the time
0054  #tempRandomValue.%B1 := #tempTime.NANOSECOND.%B0;
0055  #tempRandomValue.%B0 := #tempTime.NANOSECOND.%B1;
0056
0057  //adapt the calculated random number to the given number span
0058  #tempNormReal := NORM_X(MIN := #MIN_INT, VALUE := #tempRandomValue, MAX := #MAX_INT);
0059  // #LGF_RandomInt := SCALE_X(MIN := #minValue, VALUE := #tempNormReal, MAX := #maxValue);
0060  #LGF_RandomInt := REAL_TO_INT(#tempNormReal * (#maxValue - #minValue) + #minValue);
0061
0062  #status := #NO_ERROR;
```

Program blocks / lib

LGF_RandomReal [FC10014]

LGF_RandomReal Properties							
General							
Name	LGF_RandomReal	**Number**	10014	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version		**User-defined ID**					

Name	Data type	Default value
▼ Input		
minValue	Real	
maxValue	Real	
▼ Output		
error	Bool	
statusID	UInt	
status	Word	
InOut		
▼ Temp		
tempTime	DTL	
tempTimeStatus	Int	
tempRandomValue	UDInt	
tempNormReal	Real	
▼ Constant		
NO_ERROR	Word	16#0000
NO_CURRENT_JOBS	Word	16#7000
MAX_LESS_MIN	Word	16#8200
ERROR_IN_THIS_BLOCK	UInt	1
ERROR_RD_SYS_T	UInt	2
MIN_UDINT	UDInt	0
MAX_UDINT	UDInt	4294967295

Name	Data type	Default value
Return		
LGF_RandomReal	Real	

```
0001 //===========================================================================
0002 // Siemens AG
0003 // (c)Copyright 2017
0004 //---------------------------------------------------------------------------
0005 // Library: LGF (Library General Functions)
0006 // Tested with: CPU1212C DC/DC/DC FW:V4.2
0007 // Engineering: TIA Portal V14 Update 1
0008 // Restrictions: -
0009 // Requirements: PLC (S7-1200 / S7-1500)
0010 // Functionality: This function generates random numbers in defined limits
0011 // (Datatype Real)
0012 //---------------------------------------------------------------------------
0013 // Change log table:
0014 // Version Date In charge / Changes applied
0015 // 01.00.00 19.08.2015 Siemens Industry Online Support
0016 // First released version
0017 // 01.00.01 02.01.2017 Siemens Industry Online Support
0018 // Upgrade: TIA Portal V14 Update 1
0019 // 01.00.02 02.03.2017 Siemens Industry Online Support
0020 // Bugfix: FC number
0021 // 01.00.01 10.18.2021 Scale_X 替换成公式 OUT = [VALUE * (MAX - MIN)] + MIN
0022 //---------------------------------------------------------------------------
0023 //Status Codes:
0024 // 16#7000: No current jobs
0025 // 16#0000: Job finished // Note: There is no "Busy"-Status because the
0026 // block is finished within a single cycle
0027 // 16#8200: maxValue is samller then minValue
0028 //===========================================================================
0029
0030 //Set "No current job" status
0031 #error := false;
0032 #statusID := #ERROR_IN_THIS_BLOCK;
0033 #status := #NO_CURRENT_JOBS;
0034
```

```
0035 //Check if the maximal Value is less than the minimal value
0036 IF (#minValue > #maxValue) THEN
0037   #error := true;
0038   #statusID := #ERROR_IN_THIS_BLOCK;
0039   #status := #MAX_LESS_MIN;
0040   #LGF_RandomReal := 0;
0041   RETURN;
0042 END_IF;
0043
0044 //Read system time
0045 #tempTimeStatus := RD_SYS_T(#tempTime);
0046
0047 IF (#tempTimeStatus <> 0) THEN
0048   #error := true;
0049   #statusID := #ERROR_RD_SYS_T;
0050   #status := INT_TO_WORD(#tempTimeStatus);
0051   #LGF_RandomReal := 0;
0052   RETURN;
0053 END_IF;
0054
0055 //Callculate a random-start-value depending on the time
0056 #tempRandomValue.%B3 := #tempTime.NANOSECOND.%B0;
0057 #tempRandomValue.%B2 := #tempTime.NANOSECOND.%B1;
0058 #tempRandomValue.%B1 := #tempTime.NANOSECOND.%B2;
0059 #tempRandomValue.%B0 := #tempTime.NANOSECOND.%B3;
0060
0061 //adapt the calculated random number to the given number span
0062 #tempNormReal := NORM_X(MIN := #MIN_UDINT, VALUE := #tempRandomValue, MAX := #MAX_UDINT);
0063 // #LGF_RandomReal := SCALE_X(MIN := #minValue, VALUE := #tempNormReal, MAX := #maxValue);
0064 #LGF_RandomReal := #tempNormReal * (#maxValue - #minValue) + #minValue;
0065
0066 #status := #NO_ERROR;
```

Program blocks / lib

LRamp_001 [FC20401]

LRamp_001 Properties							
General							
Name	LRamp_001	**Number**	20401	**Type**	FC	**Language**	SCL
Numbering	Manual						
Information							
Title		**Author**	CISDI	**Comment**		**Family**	TECHNO
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_R_Tcyc	Real	
i_R_SetValue	Real	
i_R_Acc	Real	
▼ Output		
o_B_ParamErr	Bool	
o_B_EndRmp	Bool	
o_B_ZeroRmp	Bool	
o_B_Acc	Bool	
o_B_Dec	Bool	
o_R_Acc	Real	
▼ InOut		
io_R_OutRmp	Real	
▼ Temp		
tmp_R_Step	Real	
tmp_R_Delta	Real	
tmp_R_DeltaAbs	Real	
tmp_R_OutRmp_Pre	Real	
Constant		
▼ Return		
LRamp_001	Void	

```
0001 (*
0002 // ================================================================================
0003 // CISDI
0004 // (c)Copyright (2017) All Rights Reserved
0005 // --------------------------------------------------------------------------------
0006 // Library: (LPM)
0007 // Tested with: (S7-1500)
0008 // Engineering: TIA Portal (V13)
0009 // Restrictions:
0010 // 1.must be called in cyclic interrupt OB (eg.OB30)
0011 // Requirements: ( )
0012 // Functionality:
0013 // 1. linear ramp output to setvalue with a fixed acceleration
0014 // 2. slope of ramp output = i_R_Acc
0015 // 3. i_R_Tcyc and i_R_Acc must be >0.0
0016 // --------------------------------------------------------------------------------
0017 // Change log table:
0018 // Version Date Expert in charge Changes applied
0019 // 00.00.01 27.03.2017 002897 First released version
0020 // ================================================================================
0021 *)
0022 // parameter check
0023 IF #i_R_Tcyc <= 0.0 OR #i_R_Acc <= 0.0 THEN
0024   // parameter is error
0025   #o_B_ParamErr := 1;
0026   #io_R_OutRmp := 0.0;
0027   #o_R_Acc := 0.0;
0028   #o_B_EndRmp := 0;
0029   #o_B_ZeroRmp := 0;
0030   #o_B_Acc := 0;
0031   #o_B_Dec := 0;
0032   RETURN;
0033 ELSE
0034   // set ramp output to setvalue
0035   #tmp_R_OutRmp_Pre := #io_R_OutRmp;
0036   #tmp_R_Delta := #i_R_SetValue - #io_R_OutRmp;
0037   #tmp_R_DeltaAbs := ABS(IN := #tmp_R_Delta);
0038   #tmp_R_Step := #i_R_Acc * #i_R_Tcyc * 0.001;
0039   IF #tmp_R_DeltaAbs < #tmp_R_Step THEN
```

```
0040      #io_R_OutRmp := #i_R_SetValue;
0041    ELSIF #tmp_R_Delta >= #tmp_R_Step THEN
0042      #io_R_OutRmp := #io_R_OutRmp + #tmp_R_Step;
0043    ELSE
0044      #io_R_OutRmp := #io_R_OutRmp - #tmp_R_Step;
0045    END_IF;
0046
0047    // acceleration
0048    #o_R_Acc := (#io_R_OutRmp - #tmp_R_OutRmp_Pre) / (#i_R_Tcyc * 0.001);
0049
0050    // status of ramp
0051    #o_B_EndRmp := #io_R_OutRmp = #i_R_SetValue;
0052    #o_B_ZeroRmp := #io_R_OutRmp = 0.0;
0053    #o_B_Acc := #io_R_OutRmp > #tmp_R_OutRmp_Pre;
0054    #o_B_Dec := #io_R_OutRmp < #tmp_R_OutRmp_Pre;
0055
0056  END_IF;
0057
0058
```

Program blocks / System blocks / Program resources

LEAD_LAG [FB80]

LEAD_LAG Properties							
General							
Name	LEAD_LAG	**Number**	80	**Type**	FB	**Language**	SCL
Numbering	Automatic						
Information							
Title	LEAD_LAG	**Author**	SIMATIC	**Comment**		**Family**	CONVERT
Version	1.0	**User-defined ID**	LEAD_LAG				

Name	Data type	Default value	Retain
▼ Input			
IN	Real	0.0	Set in IDB
SAMPLE_T	Int	0	Set in IDB
▼ Output			
OUT	Real	0.0	Set in IDB
ERR_CODE	Word	16#0	Set in IDB
InOut			
▼ Static			
LD_TIME	Real	0.0	Set in IDB
GAIN	Real	0.0	Set in IDB
PREV_IN	Real	0.0	Set in IDB
PREV_OUT	Real	0.0	Set in IDB
LG_TIME	Real	0.0	Set in IDB

Program blocks / System blocks / Program resources

G7_RT_Plus_1_V6 [FC310]

G7_RT_Plus_1_V6 Properties							
General							
Name	G7_RT_Plus_1_V6	**Number**	310	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_IF_PAR	G7_IfParPlus_V6	
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
io_IL_ALARM_INST	Diag_Alarm	
io_SUP_ALARM_INST	Diag_Alarm	
io_IEC_TIMER_INST	TON_TIME	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

Diag_Alarm [FB701]

Diag_Alarm Properties							
General							
Name	Diag_Alarm	**Number**	701	**Type**	FB	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Base	ALARM_BASE		
Input			
Output			
InOut			
Static			
▼ Input			
SIG	Bool	false	Non-retain
TIMESTAMP	LDT	LDT#1970-01-01-00:00:00	Non-retain
SD_1	Variant		
SD_2	Variant		
SD_3	Variant		
SD_4	Variant		
SD_5	Variant		
SD_6	Variant		
SD_7	Variant		
SD_8	Variant		
SD_9	Variant		
SD_10	Variant		
▼ Output			
Error	Bool	false	Non-retain
Status	Word	16#0	Non-retain

Name	Data type	Default value	Retain
InOut			
▼ Static			
SD_0	AssocValue_0_0102		Non-retain
INIT_FB_State	DWord	16#1	Non-retain

Program blocks / System blocks / Program resources

G7_RT_Plus_2_V6 [FC311]

G7_RT_Plus_2_V6 Properties							
General							
Name	G7_RT_Plus_2_V6	**Number**	311	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
i_StepSkipping	Bool	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_SUB_1_V6 [FC316]

G7_RT_Plus_SUB_1_V6 Properties							
General							
Name	G7_RT_Plus_SUB_1_V6	**Number**	316	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_G7S	Array[0..249] of G7_StepPlus_V6	
i_indexArrayOffset	DInt	
i_ReorgArt	USInt	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_SUB_2_V6 [FC317]

G7_RT_Plus_SUB_2_V6 Properties							
General							
Name	G7_RT_Plus_SUB_2_V6	**Number**	317	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_RT_DATA	G7_RTDataPlus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
i_StepNumber	Int	
Output		
InOut		
▼ Return		
Ret_Val	USInt	

Program blocks / System blocks / Program resources

G7_RT_Plus_3_V6 [FC312]

G7_RT_Plus_3_V6 Properties							
General							
Name	G7_RT_Plus_3_V6	**Number**	312	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_IF_PAR	G7_IfParPlus_V6	
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_4_V6 [FC313]

G7_RT_Plus_4_V6 Properties							
General							
Name	G7_RT_Plus_4_V6	**Number**	313	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_5_V6 [FC314]

G7_RT_Plus_5_V6 Properties							
General							
Name	G7_RT_Plus_5_V6	**Number**	314	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
io_IL_ALARM_INST	Diag_Alarm	
io_SUP_ALARM_INST	Diag_Alarm	
io_STEP_TIME_ALARM_INST	Diag_Alarm	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_SUB_3_V6 [FC318]

G7_RT_Plus_SUB_3_V6 Properties							
General							
Name	G7_RT_Plus_SUB_3_V6	**Number**	318	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
i_indexArrayOffset	DInt	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
io_IL_ALARM_INST	Diag_Alarm	
io_SUP_ALARM_INST	Diag_Alarm	
io_STEP_TIME_ALARM_INST	Diag_Alarm	
▼ Return		
Ret_Val	Void	

Program blocks / System blocks / Program resources

G7_RT_Plus_6_V6 [FC315]

G7_RT_Plus_6_V6 Properties							
General							
Name	G7_RT_Plus_6_V6	**Number**	315	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
i_IF_PAR	G7_IfParPlus_V6	
i_RT_DATA	G7_RTDataPlus_V6	
i_G7T	Array[0..249] of G7_Transition-Plus_V6	
i_G7S	Array[0..249] of G7_StepPlus_V6	
Output		
▼ InOut		
io_G7Arrays	Array[0..13400] of USInt	
io_HMIInfo	Array[0..257] of UInt	
▼ Return		
Ret_Val	Void	

Program blocks / ctrl_synch / model

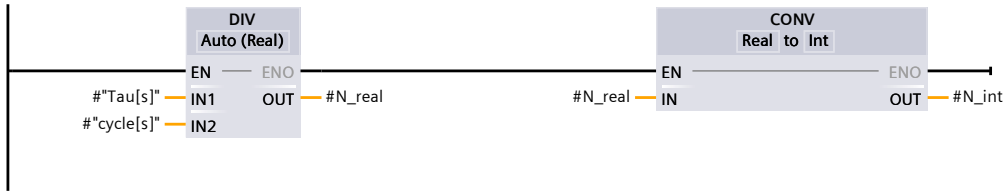
cylinder [FB4]

cylinder Properties							
General							
Name	cylinder	**Number**	4	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title	hydraulic cylinder	**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

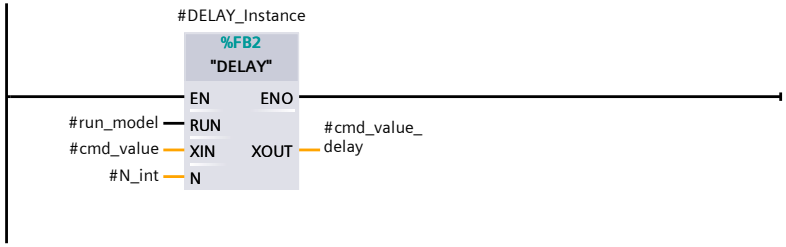
Name	Data type	Default value	Retain
▼ Input			
run_model	Bool	false	Non-retain
cmd_value	Real	0.0	Non-retain
K	Real	0.0	Non-retain
Tau[s]	Real	0.0	Non-retain
T[s]	Real	0.0	Non-retain
cycle[s]	Real	0.0	Non-retain
init_pos	Real	0.0	Non-retain
min_pos	Real	0.0	Non-retain
max_pos	Real	2000.0	Non-retain
▼ Output			
velocity	Real	0.0	Non-retain
position	Real	0.0	Non-retain
InOut			
▼ Static			
DELAY_Instance	"DELAY"		
LAG1_Instance	"LAG1"		
INTEGRAL_Instance	"INTEGRAL"		
▼ Temp			
N_real	Real		
N_int	Int		

Name	Data type	Default value	Retain
cmd_value_delay	Real		
flow_temp	Real		
velocity_temp	Real		
pos_temp	Real		
Constant			

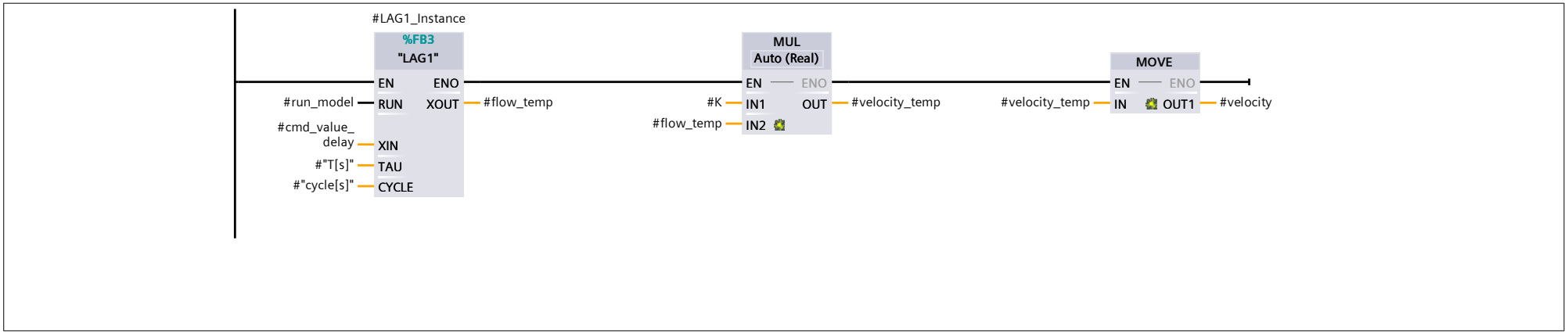
Network 1: Tau



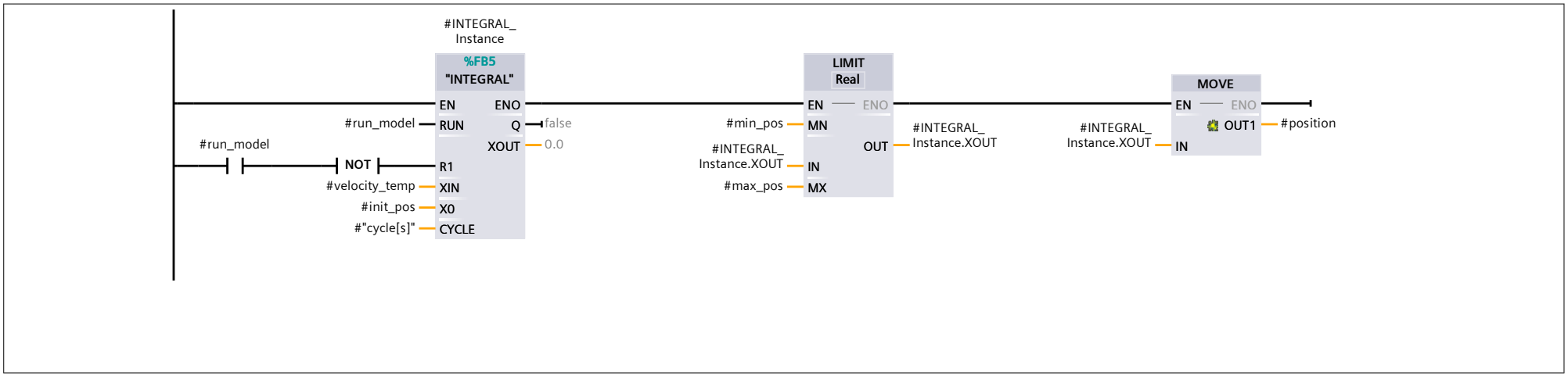
Network 2: delay



Network 3: 0.5/(Ts+1)



Network 4: position



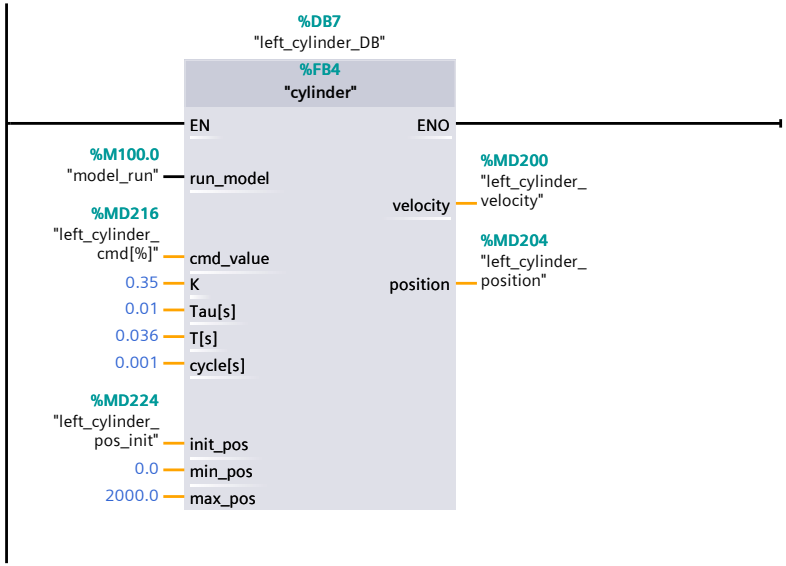
Program blocks / ctrl_synch / model

Cyclic_1ms_model [OB30]

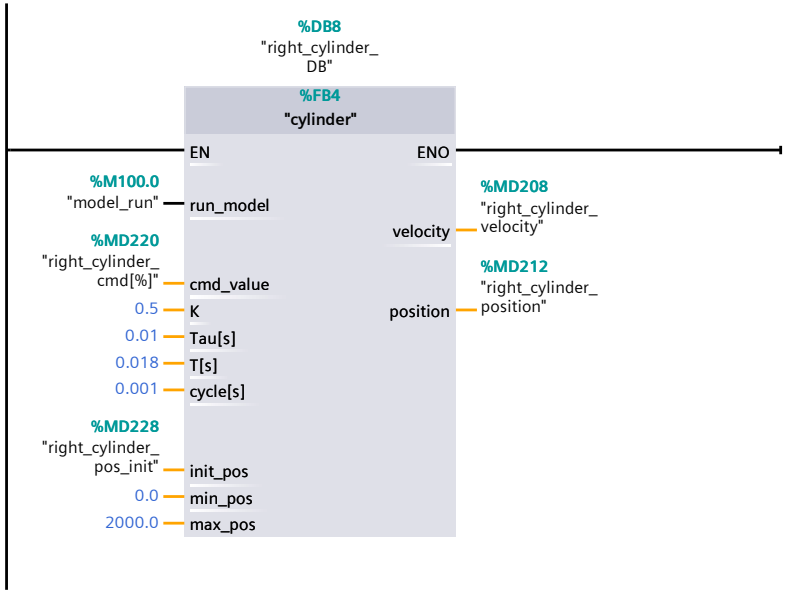
Cyclic_1ms_model Properties							
General							
Name	Cyclic_1ms_model	**Number**	30	**Type**	OB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
Initial_Call	Bool	
Event_Count	Int	
Temp		
Constant		

Network 1:


%DB7
"left_cylinder_DB"
%FB4
"cylinder"
EN
ENO
%M100.0
"model_run"
run_model
%MD200
"left_cylinder_
velocity"
velocity
%MD216
"left_cylinder_
cmd[%]"
cmd_value
%MD204
"left_cylinder_
position"
0.35
K
position
0.01
Tau[s]
0.036
T[s]
0.001
cycle[s]
%MD224
"left_cylinder_
pos_init"
init_pos
0.0
min_pos
2000.0
max_pos

Network 2:


%DB8
"right_cylinder_
DB"
%FB4
"cylinder"
EN
ENO
%M100.0
"model_run"
run_model
%MD208
"right_cylinder_
velocity"
velocity
%MD220
"right_cylinder_
cmd[%]"
cmd_value
%MD212
"right_cylinder_
position"
0.5
K
position
0.01
Tau[s]
0.018
T[s]
0.001
cycle[s]
%MD228
"right_cylinder_
pos_init"
init_pos
0.0
min_pos
2000.0
max_pos

Program blocks / ctrl_synch / model

left_cylinder_DB [DB7]

left_cylinder_DB Properties							
General							
Name	left_cylinder_DB	**Number**	7	**Type**	DB	**Language**	DB
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Input			
run_model	Bool	false	False
cmd_value	Real	0.0	False
K	Real	0.0	False
Tau[s]	Real	0.0	False
T[s]	Real	0.0	False
cycle[s]	Real	0.0	False
init_pos	Real	0.0	False
min_pos	Real	0.0	False
max_pos	Real	2000.0	False
▼ Output			
velocity	Real	0.0	False
position	Real	0.0	False
InOut			
▼ Static			
DELAY_Instance	"DELAY"		False
LAG1_Instance	"LAG1"		False
INTEGRAL_Instance	"INTEGRAL"		False

Program blocks / ctrl_synch / model

right_cylinder_DB [DB8]

right_cylinder_DB Properties							
General							
Name	right_cylinder_DB	**Number**	8	**Type**	DB	**Language**	DB
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Input			
run_model	Bool	false	False
cmd_value	Real	0.0	False
K	Real	0.0	False
Tau[s]	Real	0.0	False
T[s]	Real	0.0	False
cycle[s]	Real	0.0	False
init_pos	Real	0.0	False
min_pos	Real	0.0	False
max_pos	Real	2000.0	False
▼ Output			
velocity	Real	0.0	False
position	Real	0.0	False
InOut			
▼ Static			
DELAY_Instance	"DELAY"		False
LAG1_Instance	"LAG1"		False
INTEGRAL_Instance	"INTEGRAL"		False

Program blocks / ctrl_synch / controller

controller [FB7]

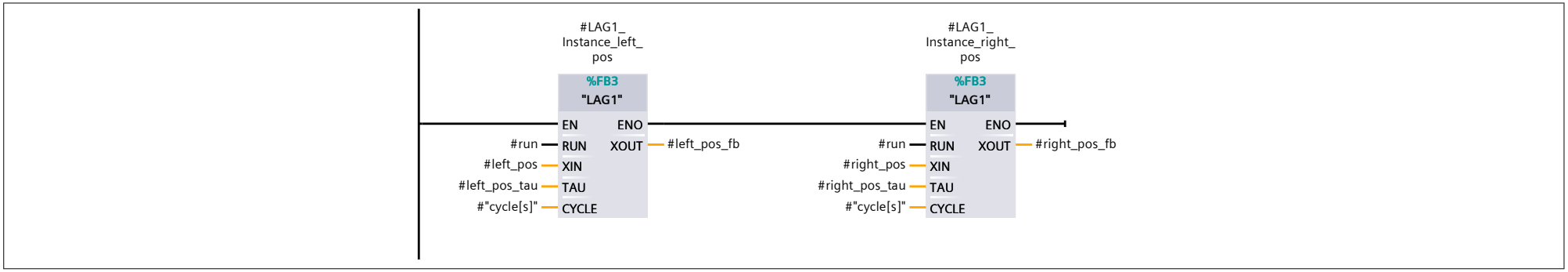
controller Properties							
General							
Name	controller	**Number**	7	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
run	Bool	false	Non-retain
ref_velocity	Real	0.0	Non-retain
ref_vel_ramp_acc	Real	0.0	Non-retain
ref_vel_min	Real	0.0	Non-retain
ref_vel_max	Real	0.0	Non-retain
left_pos	Real	0.0	Non-retain
left_pos_tau	Real	0.0	Non-retain
left_vel_fb%	Real	0.0	Non-retain
right_pos	Real	0.0	Non-retain
right_pos_tau	Real	0.0	Non-retain
right_vel_fb%	Real	0.0	Non-retain
pos_loop_dbw	Real	0.0	Non-retain
pos_loop_kp	Real	0.0	Non-retain
pos_loop_Ti[s]	Real	0.0	Non-retain
pos_loop_min	Real	0.0	Non-retain
pos_loop_max	Real	0.0	Non-retain
vel_loop_dbw	Real	0.0	Non-retain
vel_loop_kp	Real	0.0	Non-retain
vel_loop_Ti[s]	Real	0.0	Non-retain
vel_loop_ff_kp	Real	0.0	Non-retain
vel_loop_min	Real	0.0	Non-retain

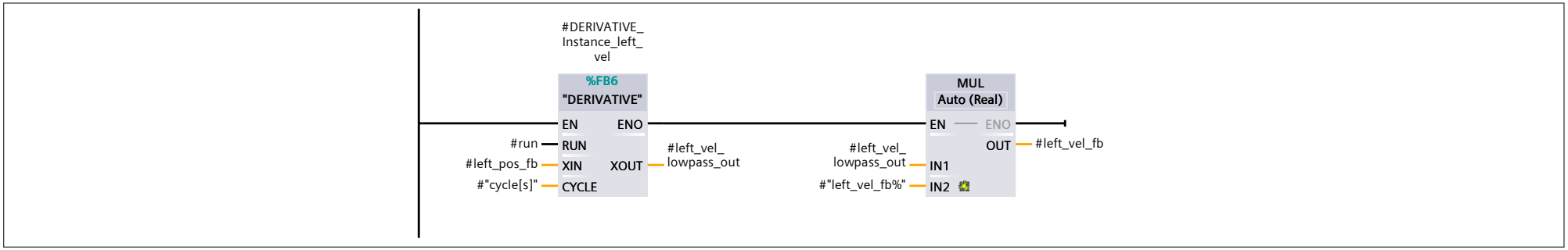
Name	Data type	Default value	Retain
vel_loop_max	Real	0.0	Non-retain
cycle[s]	Real	0.0	Non-retain
▼ Output			
ref_vel_add	Real	0.0	Non-retain
left_vel_cmd	Real	0.0	Non-retain
right_vel_cmd	Real	0.0	Non-retain
left_pos_cmd	Real	0.0	Non-retain
right_pos_cmd	Real	0.0	Non-retain
left_vel_fb	Real	0.0	Non-retain
right_vel_fb	Real	0.0	Non-retain
InOut			
▼ Static			
ref_vel_ramp	Real	0.0	Non-retain
LAG1_Instance_left_pos	"LAG1"		
LAG1_Instance_right_pos	"LAG1"		
DERIVATIVE_Instance_left_vel	"DERIVATIVE"		
DERIVATIVE_Instance_right_vel	"DERIVATIVE"		
pos_loop_Instance_left	"pos_loop"		
vel_loop_Instance_left	"vel_loop"		
pos_loop_Instance_right	"pos_loop"		
vel_loop_Instance_right	"vel_loop"		
▼ Temp			
left_pos_fb	Real		
right_pos_fb	Real		
left_vel_lowpass_out	Real		
right_vel_lowpass_out	Real		
cycle_ms	Real		
B_ParamErr	Bool		
B_EndRmp	Bool		
B_ZeroRmp	Bool		
B_Acc	Bool		
B_Dec	Bool		
R_Acc	Real		

Name	Data type	Default value	Retain
ref_pos	Real		
vel_fb	Real		
Constant			

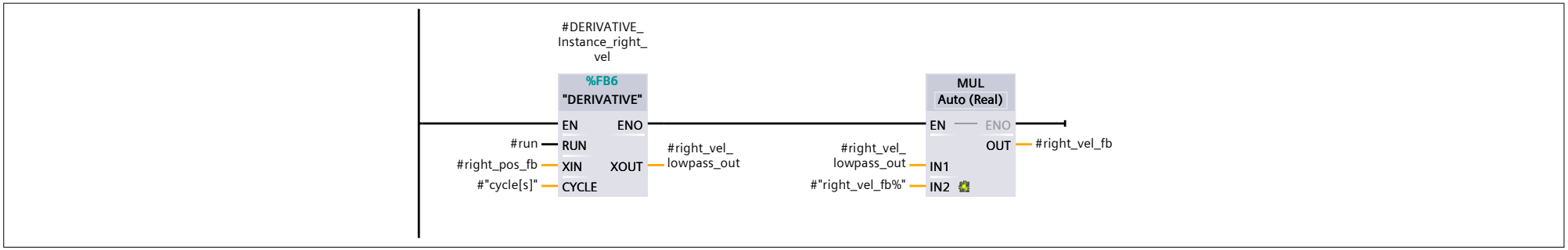
Network 1:



Network 2:

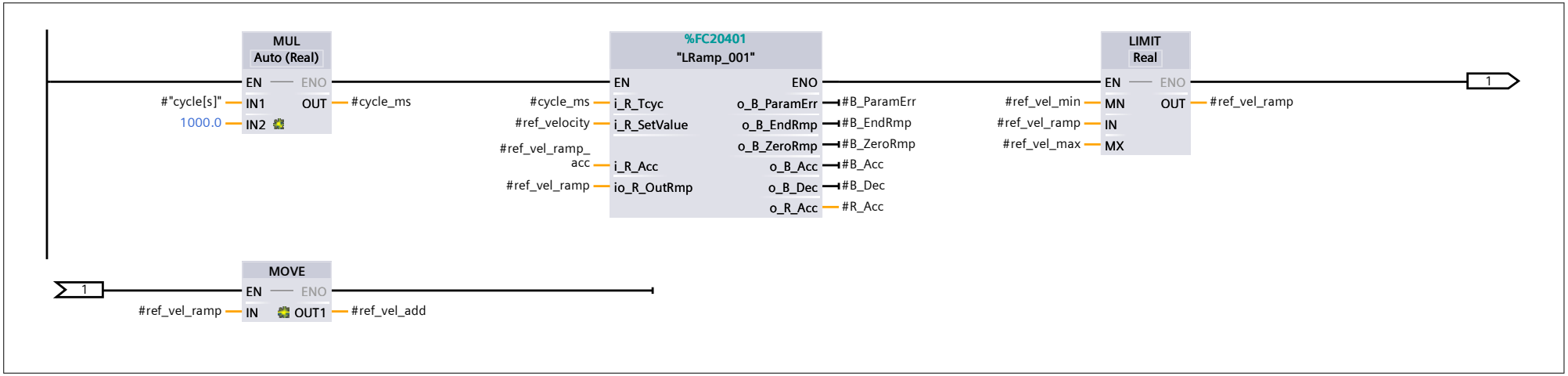


Network 3:



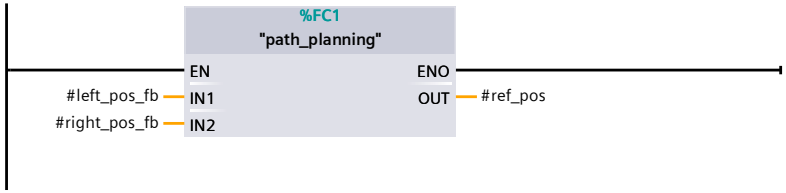
Network 4:

Network 4:



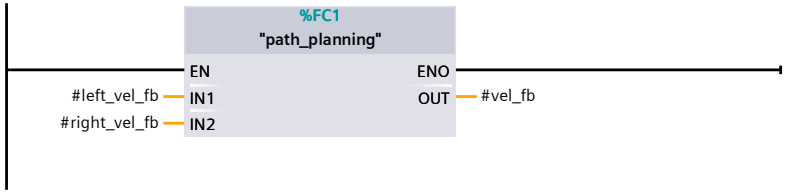
Network 5: path planning

(y1+y2)/2



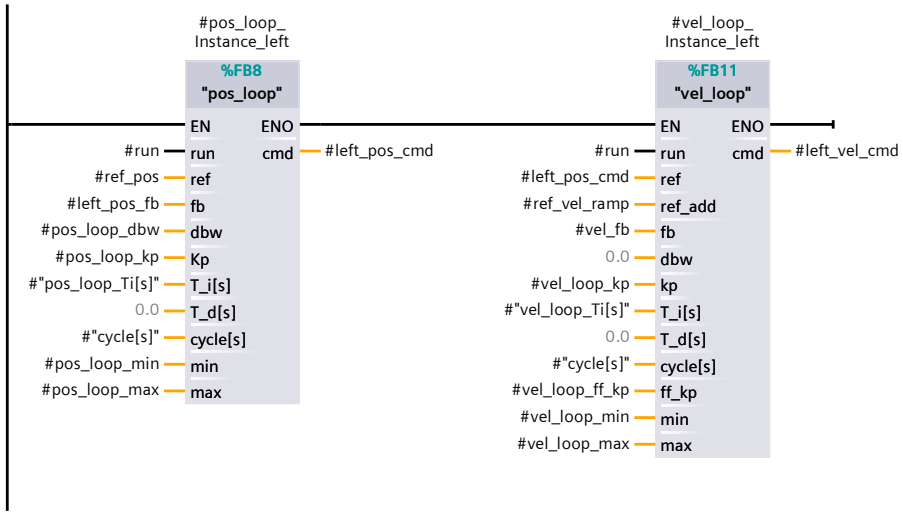
Network 6:

feedback



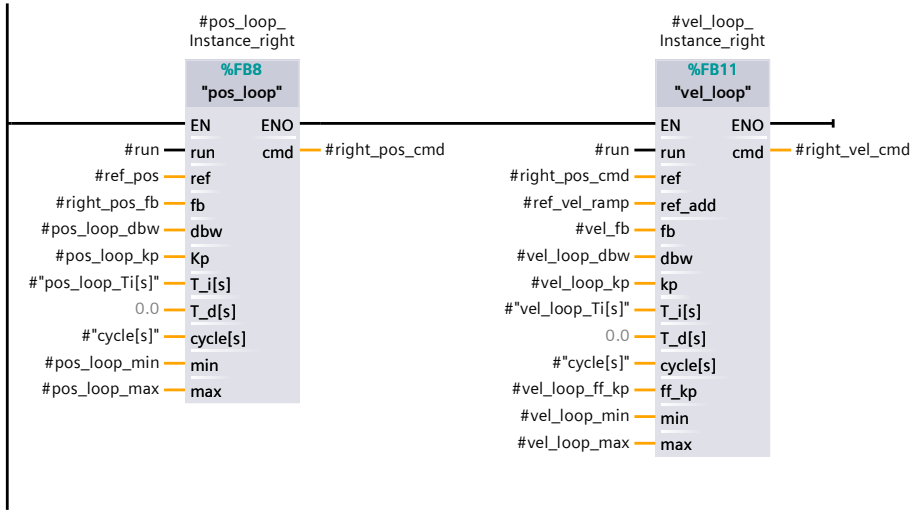
Network 7:

feedback



Network 8:

feedback velocity = (v1+v2)/2


#pos_loop_
Instance_right
%FB8
"pos_loop"
EN
ENO
#run
run
cmd
#right_pos_cmd
#ref_pos
ref
#right_pos_fb
fb
#pos_loop_dbw
dbw
#pos_loop_kp
Kp
#"pos_loop_Ti[s]"
T_i[s]
0.0
T_d[s]
#"cycle[s]"
cycle[s]
#pos_loop_min
min
#pos_loop_max
max
#vel_loop_
Instance_right
%FB11
"vel_loop"
EN
ENO
#run
run
cmd
#right_vel_cmd
#right_pos_cmd
ref
#ref_vel_ramp
ref_add
#vel_fb
fb
#vel_loop_dbw
dbw
#vel_loop_kp
kp
#"vel_loop_Ti[s]"
T_i[s]
0.0
T_d[s]
#"cycle[s]"
cycle[s]
#vel_loop_ff_kp
ff_kp
#vel_loop_min
min
#vel_loop_max
max

Network 9:


#run
MOVE
EN
ENO
0.0
IN
OUT1
#ref_vel_ramp

Program blocks / ctrl_synch / controller

path_planning [FC1]

path_planning Properties							
General							
Name	path_planning	**Number**	1	**Type**	FC	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**	energy sumption optimiza-tion	**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
IN1	Real	
IN2	Real	
▼ Output		
OUT	Real	
InOut		
Temp		
Constant		
▼ Return		
path_planning	Void	

Network 1:

```
0001 #OUT := (#IN1 + #IN2) / 2;
```

Program blocks / ctrl_synch / controller

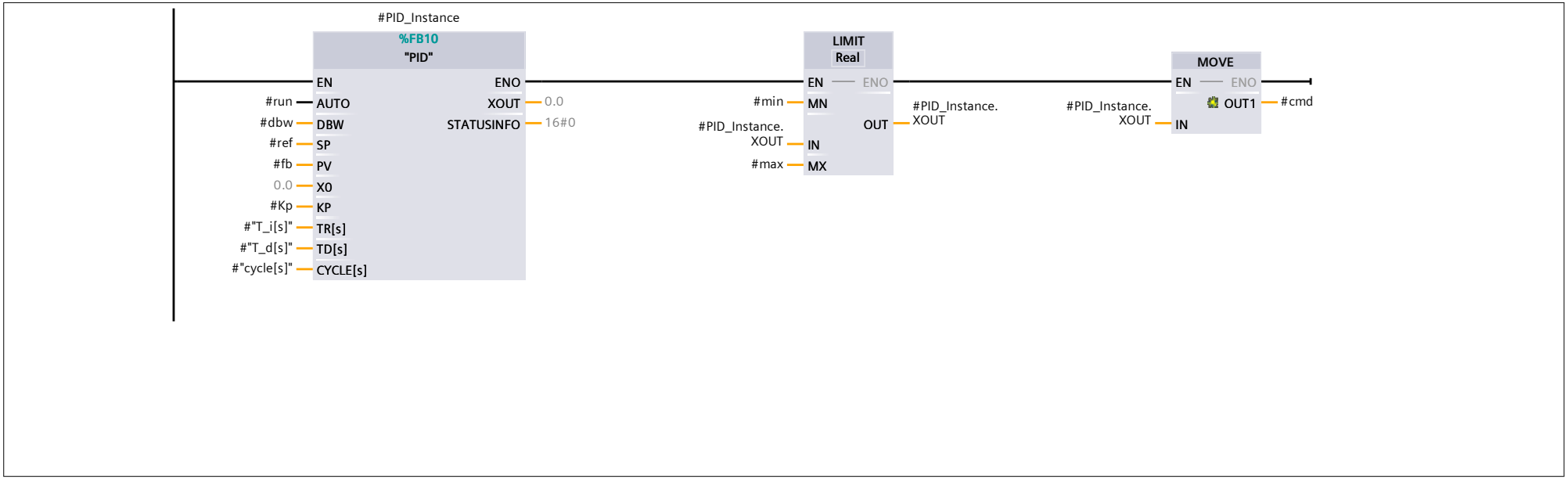
pos_loop [FB8]

pos_loop Properties							
General							
Name	pos_loop	**Number**	8	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title	position loop	**Author**		**Comment**	input:ref_pos, fb_pos output: cmd_value	**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Retain
▼ Input			
run	Bool	false	Non-retain
ref	Real	0.0	Non-retain
fb	Real	0.0	Non-retain
dbw	Real	0.0	Non-retain
Kp	Real	0.0	Non-retain
T_i[s]	Real	0.0	Non-retain
T_d[s]	Real	0.0	Non-retain
cycle[s]	Real	0.0	Non-retain
min	Real	0.0	Non-retain
max	Real	0.0	Non-retain
▼ Output			
cmd	Real	0.0	Non-retain
InOut			
▼ Static			
LEAD_LAG_Instance	LEAD_LAG		
PID_Instance	"PID"		
▼ Temp			
error	Real		
cmd_b4_out	Real		

Name	Data type	Default value	Retain
error_deadband	Real		
Constant			

Network 1:



Program blocks / ctrl_synch / controller

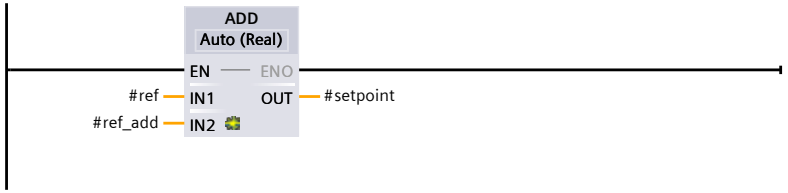
vel_loop [FB11]

vel_loop Properties							
General							
Name	vel_loop	**Number**	11	**Type**	FB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

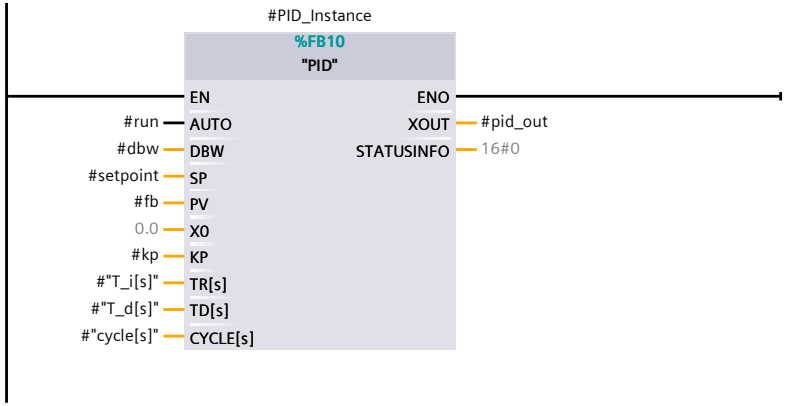
Name	Data type	Default value	Retain
▼ Input			
run	Bool	false	Non-retain
ref	Real	0.0	Non-retain
ref_add	Real	0.0	Non-retain
fb	Real	0.0	Non-retain
dbw	Real	0.0	Non-retain
kp	Real	0.0	Non-retain
T_i[s]	Real	0.0	Non-retain
T_d[s]	Real	0.0	Non-retain
cycle[s]	Real	0.0	Non-retain
ff_kp	Real	1.0	Non-retain
min	Real	0.0	Non-retain
max	Real	0.0	Non-retain
▼ Output			
cmd	Real	0.0	Non-retain
InOut			
▼ Static			
PID_Instance	"PID"		
▼ Temp			
error	Real		
setpoint	Real		

Name	Data type	Default value	Retain
pid_out	Real		
ff_out	Real		
Constant			

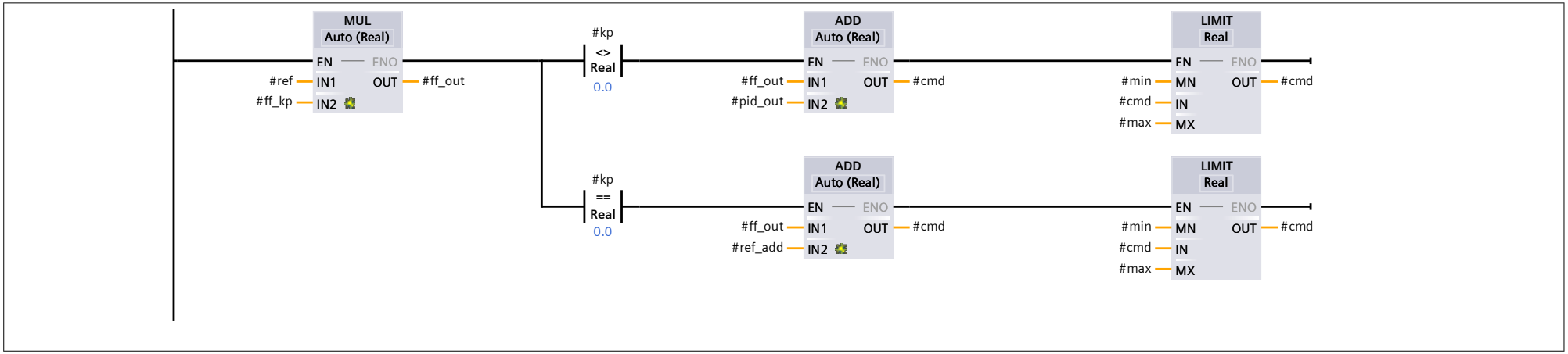
Network 1:



Network 2: velocity controller



Network 3:


MUL
Auto (Real)
EN
ENO
#ref
IN1
OUT
#ff_out
#ff_kp
IN2
#kp
<>
Real
0.0
ADD
Auto (Real)
EN
ENO
#ff_out
IN1
OUT
#cmd
#pid_out
IN2
LIMIT
Real
EN
ENO
#min
MN
OUT
#cmd
#cmd
IN
#max
MX
#kp
==
Real
0.0
ADD
Auto (Real)
EN
ENO
#ff_out
IN1
OUT
#cmd
#ref_add
IN2
LIMIT
Real
EN
ENO
#min
MN
OUT
#cmd
#cmd
IN
#max
MX

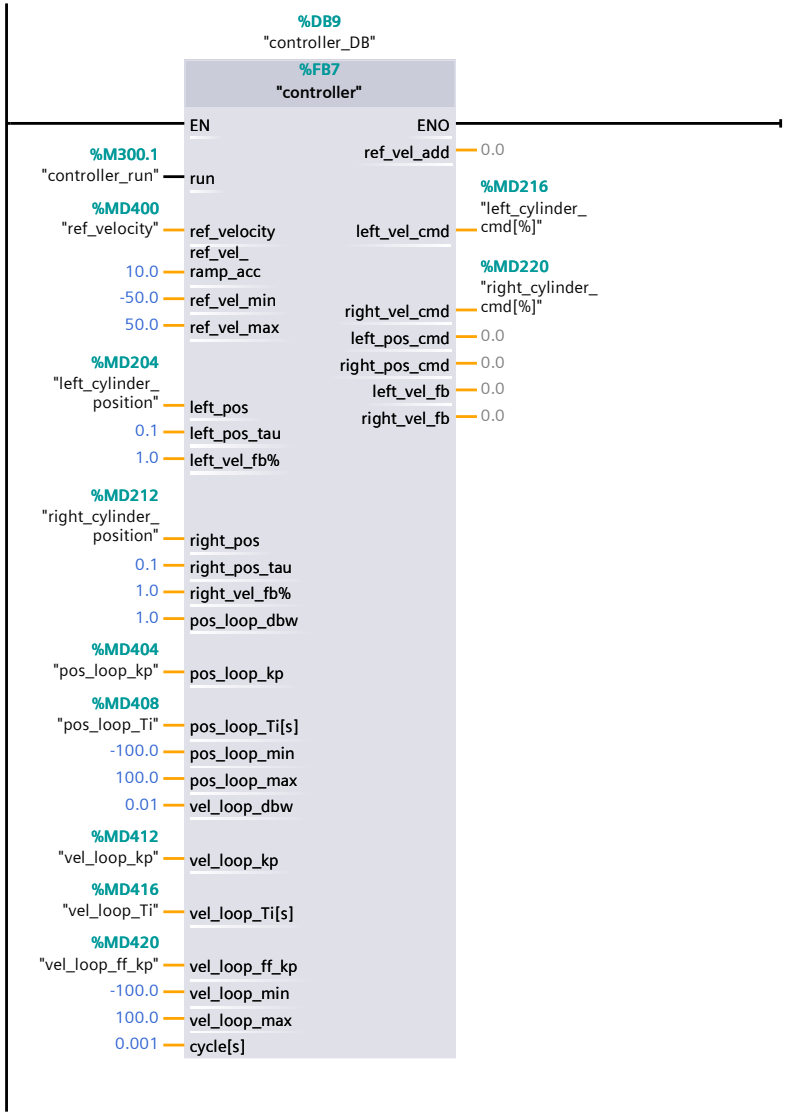
Program blocks / ctrl_synch / controller

Cyclic_1ms_controller [OB31]

Cyclic_1ms_controller Properties							
General							
Name	Cyclic_1ms_controller	**Number**	31	**Type**	OB	**Language**	LAD
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value
▼ Input		
Initial_Call	Bool	
Event_Count	Int	
Temp		
Constant		

Network 1:


%DB9
"controller_DB"
%FB7
"controller"
EN
ENO
%M300.1
"controller_run"
run
ref_vel_add
0.0
%MD216
"left_cylinder_
cmd[%]"
%MD400
"ref_velocity"
ref_velocity
left_vel_cmd
ref_vel_
10.0
ramp_acc
%MD220
"right_cylinder_
cmd[%]"
-50.0
ref_vel_min
50.0
ref_vel_max
right_vel_cmd
left_pos_cmd
0.0
%MD204
"left_cylinder_
position"
right_pos_cmd
0.0
left_vel_fb
0.0
left_pos
right_vel_fb
0.0
0.1
left_pos_tau
1.0
left_vel_fb%
%MD212
"right_cylinder_
position"
right_pos
0.1
right_pos_tau
1.0
right_vel_fb%
1.0
pos_loop_dbw
%MD404
"pos_loop_kp"
pos_loop_kp
%MD408
"pos_loop_Ti"
pos_loop_Ti[s]
-100.0
pos_loop_min
100.0
pos_loop_max
0.01
vel_loop_dbw
%MD412
"vel_loop_kp"
vel_loop_kp
%MD416
"vel_loop_Ti"
vel_loop_Ti[s]
%MD420
"vel_loop_ff_kp"
vel_loop_ff_kp
-100.0
vel_loop_min
100.0
vel_loop_max
0.001
cycle[s]

Program blocks / ctrl_synch / controller

controller_DB [DB9]

controller_DB Properties							
General							
Name	controller_DB	**Number**	9	**Type**	DB	**Language**	DB
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Start value	Retain
▼ Input			
run	Bool	false	False
ref_velocity	Real	0.0	False
ref_vel_ramp_acc	Real	0.0	False
ref_vel_min	Real	0.0	False
ref_vel_max	Real	0.0	False
left_pos	Real	0.0	False
left_pos_tau	Real	0.0	False
left_vel_fb%	Real	0.0	False
right_pos	Real	0.0	False
right_pos_tau	Real	0.0	False
right_vel_fb%	Real	0.0	False
pos_loop_dbw	Real	0.0	False
pos_loop_kp	Real	0.0	False
pos_loop_Ti[s]	Real	0.0	False
pos_loop_min	Real	0.0	False
pos_loop_max	Real	0.0	False
vel_loop_dbw	Real	0.0	False
vel_loop_kp	Real	0.0	False
vel_loop_Ti[s]	Real	0.0	False
vel_loop_ff_kp	Real	0.0	False
vel_loop_min	Real	0.0	False

Name	Data type	Start value	Retain
vel_loop_max	Real	0.0	False
cycle[s]	Real	0.0	False
▼ Output			
ref_vel_add	Real	0.0	False
left_vel_cmd	Real	0.0	False
right_vel_cmd	Real	0.0	False
left_pos_cmd	Real	0.0	False
right_pos_cmd	Real	0.0	False
left_vel_fb	Real	0.0	False
right_vel_fb	Real	0.0	False
InOut			
▼ Static			
ref_vel_ramp	Real	0.0	False
LAG1_Instance_left_pos	"LAG1"		False
LAG1_Instance_right_pos	"LAG1"		False
DERIVATIVE_Instance_left_vel	"DERIVATIVE"		False
DERIVATIVE_Instance_right_vel	"DERIVATIVE"		False
pos_loop_Instance_left	"pos_loop"		False
vel_loop_Instance_left	"vel_loop"		False
pos_loop_Instance_right	"pos_loop"		False
vel_loop_Instance_right	"vel_loop"		False