

POLITECNICO DI TORINO

Master of Science in Mechatronics Engineering



Automation and PLC-HMI Programming & Configuration

Supervisor:

Professor Luigi Mazza

Candidate:

Mohammad Shekasteh

The Academic year 2019 – 2020

In the name of God, the Most Compassionate, the Most Merciful

Thanks GOD

**Automation, PLC
Programming & HMI-
SCADA Configuration for
Lifetime Measurement of
Pneumatic Cylinder on a
Real Test Bench**

Contents

Abstract	7
Introduction	8
a) The main goal of this thesis.....	8
b) The problem of lifetime test for pneumatic cylinders	8
c) How people have studied the previous problems of this test bench.....	9
d) How this thesis would solve the previous programming problems	9
Chapter 1: Programming Software.....	11
1-1- A brief overview of automation with TIA portal software	11
1-2- Updating HSP of TIA Portal software.....	13
1-3- A brief overview of PLCSim advanced SW.....	13
1-4- Festo FluidSim software and Communication protocol.....	14
1-5- Applied components in FluidSim and FluidSim Circuit	14
1-5-1- First part of the FluidSim design is Control part	15
1-5-2- Second part of the FluidSim design is Circuit part.....	17
1-5-3- Included components on this FluidSim simulation	18
1-6- The main goal of the thesis and Description of the PLC Program.....	22
1-6-1- What is the WEAR Mode Test & the problems	22
1-6-2- What is the LEAK Mode Test & the problems	23
1-6-3- What is the BOTH Modes Test & the problems.....	24
1-6-4- TIA Portal and Creating New Project and Blocks.....	25

1-6-5-	Device configuration, Interconnections and IP addresses	28
1-6-6-	Memory configuration of PLC device	34
1-6-7-	Functionality of blocks in the PLC program	38
1-6-7-1-	General idea about the program.....	38
1-6-7-2-	How to begin programming & create OB, FB, FC or DB.....	38
1-6-7-3-	Function Blocks and Data Blocks of the Project.....	40
A. 1-7-	Functionality of whole program Blocks and Networks.....	43
1-7-1-	PLC Tag table.....	43
1-7-2-	Main [OB1]	45
1-7-2-1-	Content of “Main OB1” totally 34 networks.....	46
Network 1 to 10;	46
Network 11;	Set Priority conditions in “Both Modes” test	48
Network 12;	Finished Priority signal to start “LEAK Mode”	49
Network 13:	Activating output HPV	49
Network 14:	Activating output LPV	50
Network 15:	Activating output ABV	51
Network 16:	Activating output PBV.....	51
Network 17:	Activating inputs "Wear select Mode"	51
Network 18:	Activating inputs "Leak select Mode"	51
Network 19:	Activating inputs "Both select Mode"	52
Network 20:	Activating inputs "Door Limit switch"	52
Network 21:	Activating inputs "Emergency Button"	53

Network 22: Activating inputs "Reset Counter Master_HMI"	53
Network 23: Activating inputs "Reset Counter Each Cycle_HMI" ...	54
Network 24: Activating inputs "Reset Cycle Total_HMI"	54
Network 25: Activating inputs "Start Button"	55
Network 26: Activating inputs "Reset Cycle Total_Cabinet"	56
Network 27: Program block of "Selector Test Mode_HMI"	57
Network 28: Program block of "Indicator Light Mode_HMI"	59
Network 29: Activating output "Wear Test LED"	60
Network 30: Activating output "Leak Test LED"	61
Network 31: Activating output "Both Test LED"	61
Network 32: Activating output "Finished Test LED"	61
Network 33: Activating output "Error Test LED"	62
Network 34: Activating one memory for "Reset Button"	62
1-7-3- Test Bench Program Block_Function Block [FB1]	62
1-7-3-1- Network 1 to 14 is about "LEAK Test Mode"	63
1-7-3-2- Network 16 to 27 is about "WEAR Test Mode"	72
1-7-3-3- Network 29 to 35 is about "Both Test Mode"	83
1-7-3-4- Network 37 to 41 is about "LED Output Indicators"	87
1-7-3-5- Network 43 to 45 is about 3 different part.....	90
1-7-4- Selector Test Mode Function Block - For HMI [FB2]	93
1-7-5- The Light around Test Mode Buttons on HMI [FC1]	95
B. 1-8- HMI Conflagration and Design	96
1-8-1- Root Screen Display of HMI	98

1-8-2- Button & Indicators Screen.....	108
1-8-3- POLITO Logo Screen	109
1-8-4- POLITO Template Screen	110
C. 1-9- Analogue Data Saving in HMI Storage	111
1- Chapter 2: Electrical and hardware implementation.....	113
2-1- Physical Utilized components in this test bench	113
2-2- The implementation & Assembling of the project	126
Conclusion and Suggestion	128
Future developments and suggestions.....	128
References	129
Appendix and the Main complete PLC Program and Codes	130

Abstract

Many thank our unique GOD of the world for giving me this opportunity to do perfect research and performance on one of the best topics in Automation and Programming. This test bench mainly focuses on the Siemens PLC Programming & HMI-SCADA touch panel Configuration to control some pneumatic cylinders and actuators in order to measure their lifetime durability.

All industries are keen on knowing how long they can utilize from a component especially regarding component which has movement actions. This thesis is working on creating a creative PLC program and designing the Human Machine Interface (HMI) to display the movement of cylinders and the status of each sensor at each moment. Some components are needed for commencing the examination of the Test Bench that consists of 10 pneumatic cylinders with some common control components such as 4 main common valves & HMI display also one control valve specifically is needed for each pneumatic cylinder. The PLC program is included of one main common FB block which is used for all cylinders but with the different, IN/OUT addresses and are controlled with different variables in the input part and send the signal to the different actuators and indicators on HMI and cabinet. With this test bench, two different tests can be performed separately and manually or successively and automatically one after the other. One is called the WEAR test (doing cycling test) and the other one is called LEAK (leakage test). Usually, testing starts with the WEAR and performs 30,000 cycles and then immediately performs one LEAK on the Posterior chamber and one on the Anterior chamber and continues the process. During the Leak test, every 5 seconds the pressure is measured and sampled with an analogue manometer and is recorded in a TXT file on the HMI storage for any needed analysis. The significant point of this test bench is that if during each of the tests, a problem appears for any of cylinders or actuators, it automatically is stopped, and other cylinders keep continuing until they reach the total setpoint on the HMI. Two main setpoints can be set on the display, one for each period cycle and the other one is for the total cycle which should be set by the operator. Final saved data can be transferred to a PC by the attached USB drive from the back of HMI.

Introduction

- a) The main goal of this thesis
- b) The problem of lifetime test for pneumatic cylinders
- c) How people have studied the existing previous problems of this test Bench at Polito University, DIMEAS Lab department.
- d) How this thesis is going to solve the previous programming problem

a) The main goal of this thesis

The main goal is the measurement of the pneumatic cylinders' working hours that can help the industry to be used and improve. So, when we could find a solution with automation and programming with a logic controller like PLC devices to run Pneumatic cylinders for a certain number of cycles and put them under a certain pressure to measure the lifetime of any cylinder.

For executing, for this kind of test, we need to have an automated program to run the test for a long time. This type of automatic operation can be performed by a different variety of software and devices which depends on the specialist of people and/or the different environments and availability of controllers such as Arduino, Microcontrollers, PLC devices, and other different logic controllers' program.

In this case here, I will explain, design and investigate the measurement with a PLC device from SIEMENS company.

The designing, programming, and implementation of the whole test bench have been performed completely practical by Eng. MOHAMMAD SHEKASTEH.

b) The problem of lifetime test for pneumatic cylinders

In the case of testing pneumatic cylinders, they should be tested into two different cases, the first is the LEAK Test and the second one is the WEAR test. Both tests

are used to examine and measure the durability and the tolerance of the anterior and posterior chambers of the pneumatic cylinders, exactly the ring part which piston moves through them in and out.

To perform this test without automated electric and pneumatic circuits would be very difficult and wasting time and energy, hence we need to design an operating circuit that can be executed for these tests in an automated way.

c) How people have studied the previous problems of this test bench

Some students have already worked on this project with a different old version of PLC, S7-200 with S7 MicroWin software, which there were many limitations for designing the program at POLITO University, DIMEAS Lab. However, they have done it in a good way with the available instruments. One of the problems was the limitation of the software and lack of the different useful functions and different toolboxes. Besides, the next problem was the volume of the hardware and wiring up on the test bench. Next to all, I can mention that controlling and operating the old test bench with this was difficult. Also, saving data in memory has many limitations of the storage volume to keep the data for a long time. But these kinds of difficulties pretty have been solved by the updated PLC Automation software and HMI abilities.

d) How this thesis would solve the previous programming problems

The main problem of that programming can be about limitation in the usage of the function in the software plus saving data in external memory in a computer by a cable connection. By using the updated software, which is called TIA Portal, this problem has been solved. There are many useful functions in this software that reduce the volume of the program and give us more flexibility and more functionality in designing a circuit. Some other comfortabilities come to us by

connecting the HMI touch panel to our SIMATIC S7-1500 PLC. It can be more controllable by using a touch panel interface with different pages and/or a variety of layers.

On this test bench, many more cylinders are controlled by a very shorter and categorized program. Besides, as I mentioned by using an HMI, we can control and display all cylinders and actuators, so we can remove the physical switch pad from the test bench.

Chapter 1: Programming Software

- 1-1- A brief overview of automation with TIA portal software
- 1-2- Updating HSP of TIA Portal software
- 1-3- A brief overview of PLCSim advanced SW
- 1-4- Overview of Festo FluidSim software and Communication protocol
- 1-5- Applied components in FluidSim and Circuit
- 1-6- The main program in TIA Portal
- 1-7- The project needs to be structured and implemented
- 1-8- Programming / Diagrams / HMI Configuration / Ladder diagrams / Saving data

Programming Software

1-1- A brief overview of automation with TIA portal software

There are many different types of controllers, with different brands, to use for automating processes. For this test bench, it has been preferred to use the SIEMENS controller, PLC S7-1500.

The SIEMENS automatic system devices are used in many projects from the subsystems of an automation solution under a uniform system construction to form a homogeneous all from the slave level up to the master level.

The **TIA** which stands for the “**Totally Integrated Automation**” concept allows using uniformly all automation components and instrumentations which is using a single system platform and tools with simultaneous operator interfaces. These requirements are fulfilled by the SIMATIC automation system, which provides

uniformity for programming, configuration, communication and data management.

In this thesis, it is used the newly developed SIMATIC S7-1512 PLC device plus Basic Panel **HMI** which stands for “**Human Machine Interface**”.

STEP 7 Professional 1500 controllers are compact in designing and it could be expanded by extra modules. The CPUs feature has integrated by bus interfaces for communicating with other automation systems via Industrial Ethernet cable and, depending on the type of module, via PROFIBUS interface as well. This is a very common tool for the configuration of the hardware and generation of the user program.

➤ *There are five different programming languages for producing the user PLC program:*

a. (LAD):

A graphic representing the Ladder logic diagram (LAD) which is very similar to an electric circuit diagram.

b. (FBD):

Function Block Diagram (FBD) is also a graphic representation, but it is based on electronic circuit and logic gates.

c. (SCL):

Structured Control Language (SCL) is high-level programming.

d. (STL):

Structured Text Language (STL) is another one that is using the formulation of the control system as a list of different commands.

e. (GRAPH)

A sequencer language is a GRAPH as a sequential processing program.

It supports testing of the user program, using watch tables for monitoring by using control and forcing the value of tags, by executing the program with the user Tag values during run operation, and by offline simulation of the PLC.

1-2- Updating HSP of TIA Portal software

HSP is the abbreviation for the **Hardware Support Package**. Updating the hardware catalogue of the TIA Portal by installing the Hardware Support Packages (HSPs). It consists of the software components for integrating a device with a variety of associated firmware versions into the hardware catalogue of the TIA Portal. Depending on the version of the TIA Portal the devices are already integrated or are missing. The HSPs are specific to the versions of the TIA Portal. The version of the TIA Portal must correspond at least to the version of the HSP. HSPs have the file name extension “.isp [version number]”.

1-3- A brief overview of PLCSim advanced SW

A virtual controller which is called Simatic S7-PLCSim advanced is to simulate any PLC program which is created by Siemens software products. It is used for a comprehensive simulation of any designed PLC programming. PLCSim advanced mainly is used for virtual communication and simulation to connect a virtual controller to one virtual model for a machine or device to high-level language programming applications. One of the most significant benefits of this virtual machine can be testing of the PLC program in TIA Portal software and convenient testing for user program with the simulation tables and PLCSim without having a real PLC. It is noted that any different version of the TIA portal might require a different version of PLCSim Advanced, but usually, the last versions of the TIA

portal can be supported by PLCSim Advanced version 2.0. The link for more details about Function Manual of PLCSim Advanced, SIMATIC S7-1500 is mentioned here;

https://cache.industry.siemens.com/dl/files/153/109739153/att_895955/v1/s7-plcsim_advanced_function_manual_en-US_en-US.pdf

1-4- Festo FluidSim software and Communication protocol

For simulating pneumatic, hydraulic and electrical circuits, FluidSim is a leading software to execute them in a very nice way. For the study of electro-pneumatic, designing electronic and digital circuits and simulation, creating and structural design can be used and would be the most common and comprehensive software. For simulating, it is necessary to select the “*OPC mode*” from the tab “*options*” on the top of the FluidSim software, under the menu of “*Easy port/OPC/DDE connection*”. Then in the FluidSim software, after pushing the play button to start simulating the window “*Festo Didactic EzOPC*” would be opened that it must be set. Click one of the tabs on the top of this window Which is called “*S7-PLCSIM*”. It should be put the checkmark in the small checkbox of “*PLCSIM Advanced*”. However, this window can show some other states of the connection in the other tabs. Although in the TIA portal, some setting should be done too like in the FluidSim.

1-5- Applied components in FluidSim and FluidSim Circuit

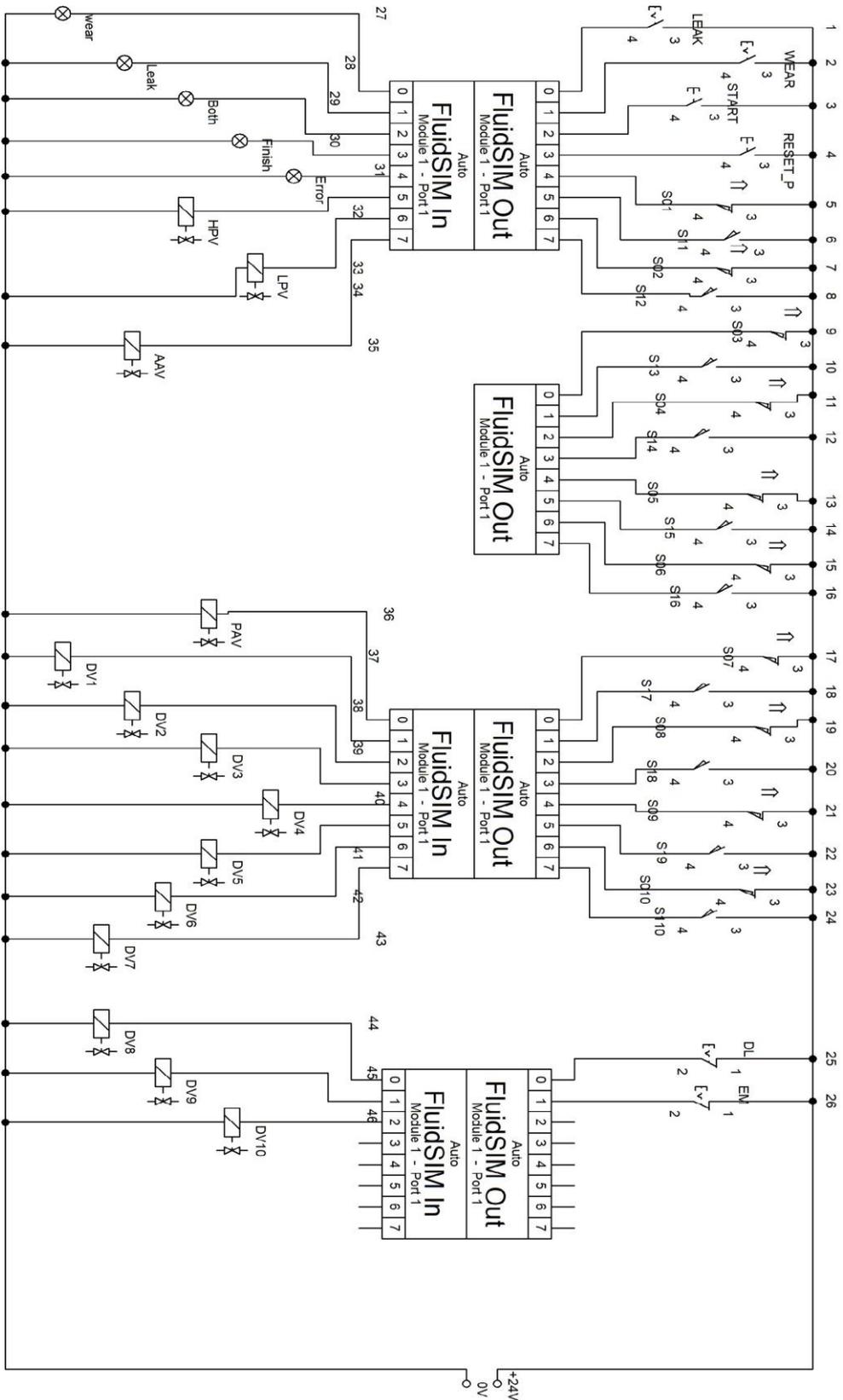
In this circuit, some electric and pneumatic components have been used that most of them have a monostable status and few of them have bistable status. It means all cylinders, valves and blocking valves are monostable.

The whole circuit is consisting of two main designs in the same project. The first design part is about input and output control and indicators that are connected to

the input-output blocks between the PLC program and FluidSim. Regarding the second part of the design that is the figure of all Cylinders, valves and all interconnections between them.

1-5-1- First part of the FluidSim design is Control part

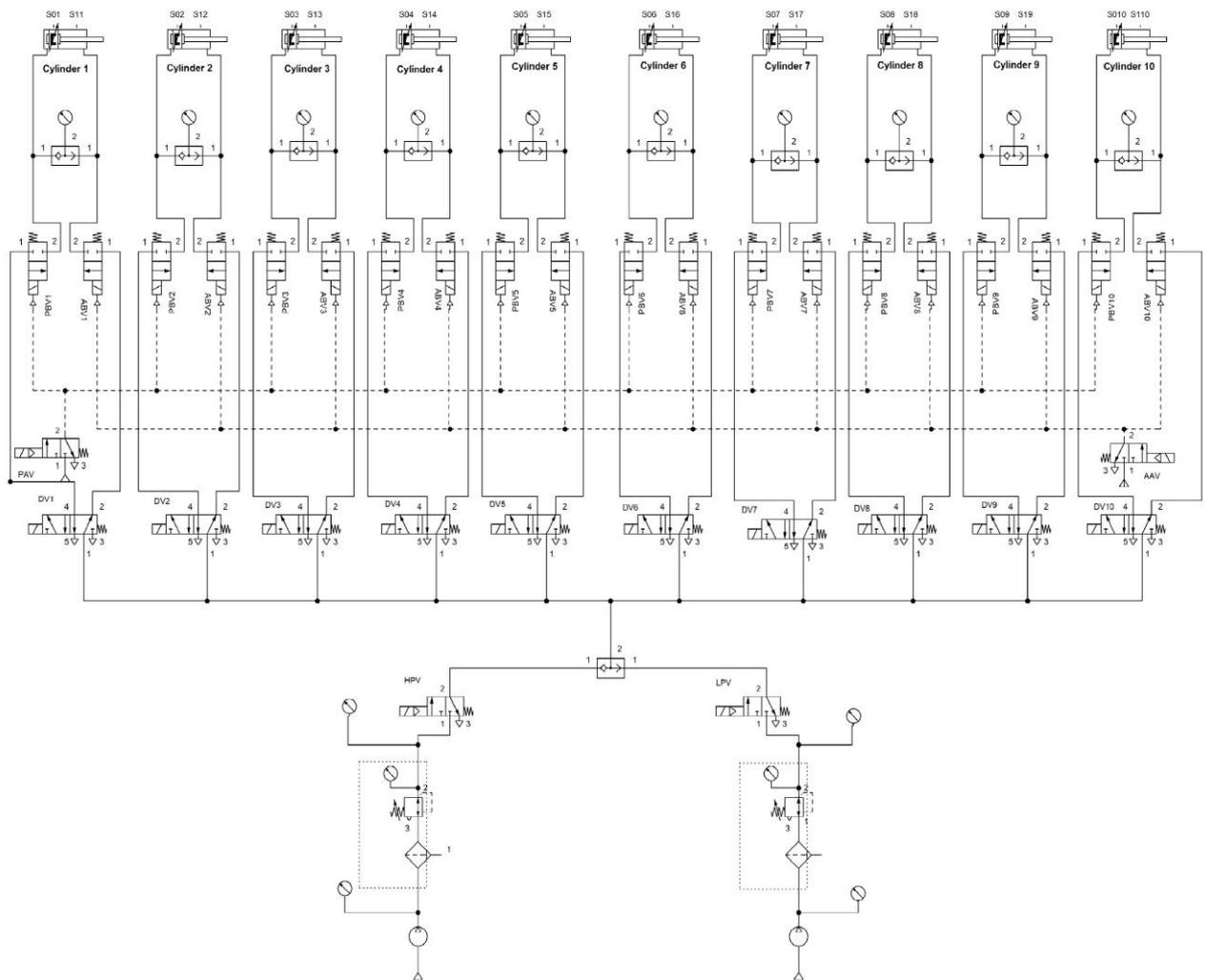
The control part of the FluidSim circuit which is connected to virtual PLC signals are consists of all input and output and they are shown by LED indicators for outputs and some pushbuttons as inputs signal. The input and output blocks connect the virtual internal valves to the I/O of the main PLC program by using PLCSim advanced software.



Figure

1-5-2- Second part of the FluidSim design is Circuit part

The photo below it is showing all components and the type of connections between them which dash lines are showing pneumatic control signals and the lines are showing air commands.



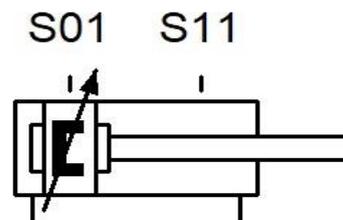
Figure

1-5-3- Included components on this FluidSim simulation

a) Ten double-acting cylinders;

These cylinders must be sent the pneumatic pressure command to one of the back or front chamber for moving the Cylinder OUT or IN.

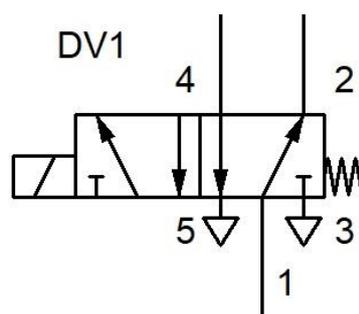
In the figure below a double-acting cylinder is shown.



Figure

b) Ten Electro-pneumatic distribution valves 5/2

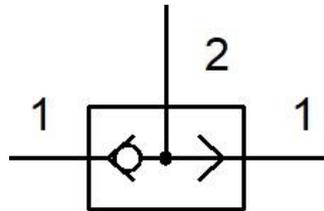
These distribution valves are the main valves to control the cylinders to move out or move in by receiving an electrical signal. So, for controlling each cylinder there must be its own valve with a separated electric control signal which arrives directly from PLC.



Figure

c) Eleven Pneumatic OR gates

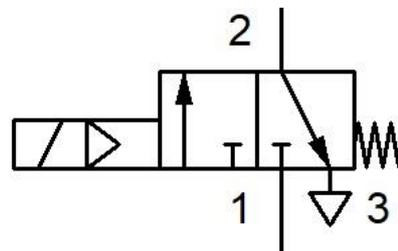
About ten of these pneumatic OR gates have been used for Connecting the manometer to the anterior chamber and posterior chamber of each cylinder to allow manometer to have access to both chambers.



Figure

d) Two main electro-pneumatic Feeding valves 3/2

There are two main valves that control entrance air to the circuit. One of them is used for high-pressure air during the LEAK test and the other one is used for low-pressure air during the WEAR test. It is important that they should not be energized at the same time.

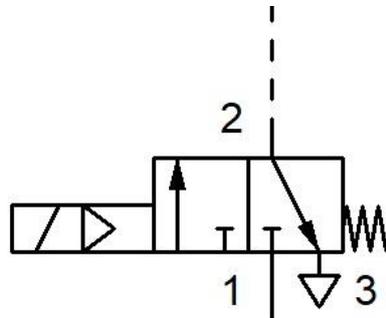


Figure

e) Two main electro-pneumatic control valves

For controlling all blocking valves which are located at the path of the posterior chamber and anterior chamber, these two main pneumatic valves control.

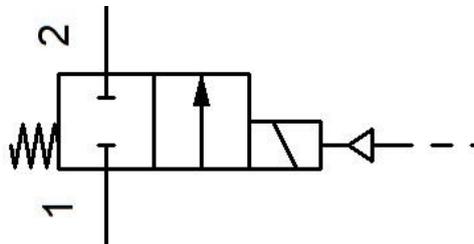
One of the pneumatic control valves sends a pneumatic command to ten anterior chambers to block the front path and the second one sends a command to ten posterior chambers to block the posterior path of all cylinders in the same time.



Figure

f) Twenty pneumatic blocking valves

There is one pneumatic blocking valve on the path of each chamber for all cylinders and all chambers to block or allow passing the air. These are located Exactly before the input of each chamber.



Figure

g) Twenty Reed sensors

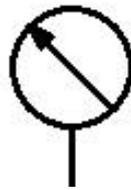
These sensors are sensitive to the magnetic fields and there is a magnetic field on the moving bottom of each stroke inside the cylinders which is sensed by this sensor easy. it sends the signal to the PLC by only 2 wires.

A REED sensor is an electrical contact used in a magnetically operated switch or relay.

The figure of these sensors is shown as S0 and S1 over the double-acting cylinder figure in the first item of this section.

h) Ten analog pressure sensors

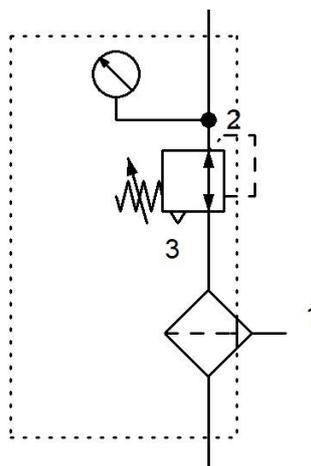
Pressure Manometers are used for measuring the pressure of each chamber. One pressure Manometer is used for both chambers in common which is connected to both chambers by an OR logic gate.



Figure

i) Two pressure regulators

There are two separate pressure regulators in the main feeding inputs for high pressure and low pressure which are connected to the two main input valves.



Figure

1-6- The main goal of the thesis and Description of the PLC Program

When we use a cylinder in our system, we need to know how long it can work and how long is its durability. So, while we have this kind of information about the cylinders it would be useful to estimate the expenses of the system. In this case, it can be tested from point of view of the duration time which is called the WEAR test and for another case, from point of view of the pressure and load that is called the LEAK test. In another word, for the testing lifetime of a cylinder, there are two different kinds of tests that allow measuring the working time hours of each cylinder under a load and during cycling for each cylinder that respectively the first one is called WEAR test and the second one is called LEAK test.

1-6-1- What is the WEAR Mode Test & the problems

In the case of testing cylinders WEAR means cycling the cylinders forward and backward. For doing this test, it needs to measure the maximum number of done cycling for each cylinder, one cycle means to extract and retract the cylinder out and in. For doing this test we have to reach a maximum number of cycles which is called “*Cycle Setpoint Each*” for each cylinder. Usually, the maximum cycling number for this test can be set 30,000 cycles. It means when the Counter reaches this setpoint, the test bench will be stopped.

There might appear different kinds of problems for doing and measuring this test. Piston rod might be in these three positions. the first one is the piston rod cannot reach the end of course by 4 s, because of the length of the stroke which is 250 mm, during forward-moving or backward moving. The second position is that the cylinder remains inside due to a lack of sensor or sensor problems. The last position is the cylinder extracted due to a lack of sensor or any sensor problems. All these position statuses would be considered in the PLC program.

1-6-2- What is the LEAK Mode Test & the problems

For measuring and testing the resistance and tolerance of a cylinder it needs to be done at the LEAK test and any leakage can be measured easily. It consists of two tests first one is a test over the front chamber and the next one is a test over the back chamber. In other words, for doing one LEAK test, each chamber must be pressurized once separately, for 30 minutes.

- The first part of the test is in the Anterior chamber

In this test when the piston is located inside it should be filled up the front chamber by air pressure, although there are some important steps that all paths over front and back chamber must be open to allow passing the airflow by energizing the High-Pressure Valve and Blocking Valves but not Distribution Valve at this moment. After pressurizing the chamber for 10 s, both front and back paths must be blocked for 60 s, it means both blocking valves must be de-energized for closing the cylinders' chambers, and then, after 5 s supply pressure must be cut and disconnected by input High-Pressure Valve (HPV). And after 5 seconds the Posterior Blocking Valve over the front path must be energized again to open the path through the exhaust, and Distribution valve It should be open in order to pass any possible leakage in the front chamber. It means that the Distribution Valve (DV) must not be energized because it already keeps opening the back path. The duration of the last step of the test must be a minimum of 30 minutes.

- The second part of the test is in the Posterior chamber

The state of the testing posterior chamber is, while the cylinder has been moved out and is located at the extracted position. Testing this chamber is a little like the previous chamber with some steps. In the beginning, all the paths to the front and back chamber must be open. So, in this case, there is only one difference which is the state of the Distribution Valve that also should be energized along with High-Pressure Valve and Blocking valves to keep the piston extracted for 10 s to be pressurized. Then by de-

energizing only Blocking Valves the front and back paths to the cylinder will be blocked and remain in this state for 60 s. After this time, the Feeding supply must be disconnected by up High-Pressure Valve. Now for starting the test, only Anterior Block Valve should be energized and Distribution Valve de-energized after 5 s to allow the airflow passes through the exhaust if there is any possible leakage in the posterior chamber. The duration of the last step of the test must be a minimum of 30 minutes and then the test bench will be stopped.

1-6-3- What is the BOTH Modes Test & the problems

In this mode, both tests are continuously working to reach a maximum number of cycles which is called “*Total Setpoint*”. It mentioned in previous sections, there are two main tests LEAK and WEAR tests. It means there are two different modes plus another mixed common mode. This Mode consists of a WEAR and a LEAK mode that is called “BOTH Modes”. By running Both Modes, the test bench will start to run first WEAR test and after reaching the Cycle Setpoint, it automatically will swap to LEAK test and does the LEAK test in front chamber and then back chamber and again it would return to perform the WEAR test and it would be happening successively and continuously performing until reaching the “total setpoint” cycle. It is very important to give the priority performance to the cylinder that is working in WEAR Mode. It means when all cylinders finish the WEAR test then they can perform the LEAK test at the same time.

There are some issues during the “BOTH Modes” test. First, all cylinders must finish in the WEAR test and then all cylinders at the same time together can swap to the LEAK test. This should be performed because in two different modes there are two main feeding valves with different pressures. One of them with 6 bars for implementing the WEAR test and the next one is with 10 bar for implementing the LEAK test. In addition, giving priority for performing each test successively without any interruption. In this case, we need to send two pulses to those different

modes to run them automatically. Another one is a little important and a little more complicated, pushing the emergency in the different steps of the LEAK test. when we push the emergency push button in each step, different valves should be set or reset that is confusing.

1-6-4- TIA Portal and Creating New Project and Blocks

One of the best software for automation programming is the TIA portal which has been produced by SIEMENS company. TIA stands for Totally Integrated Automation. There are a variety of versions on this program but in this case, I have used the latest version which is 15.1 and it was the only one that I could use because the firmware version of the real provided PLC on this test bench was supported by the last version of TIA portal SW. The version of the firmware was 2.6. Although It was required to install hardware support package HSP for another component in this test bench which is remote IO control.

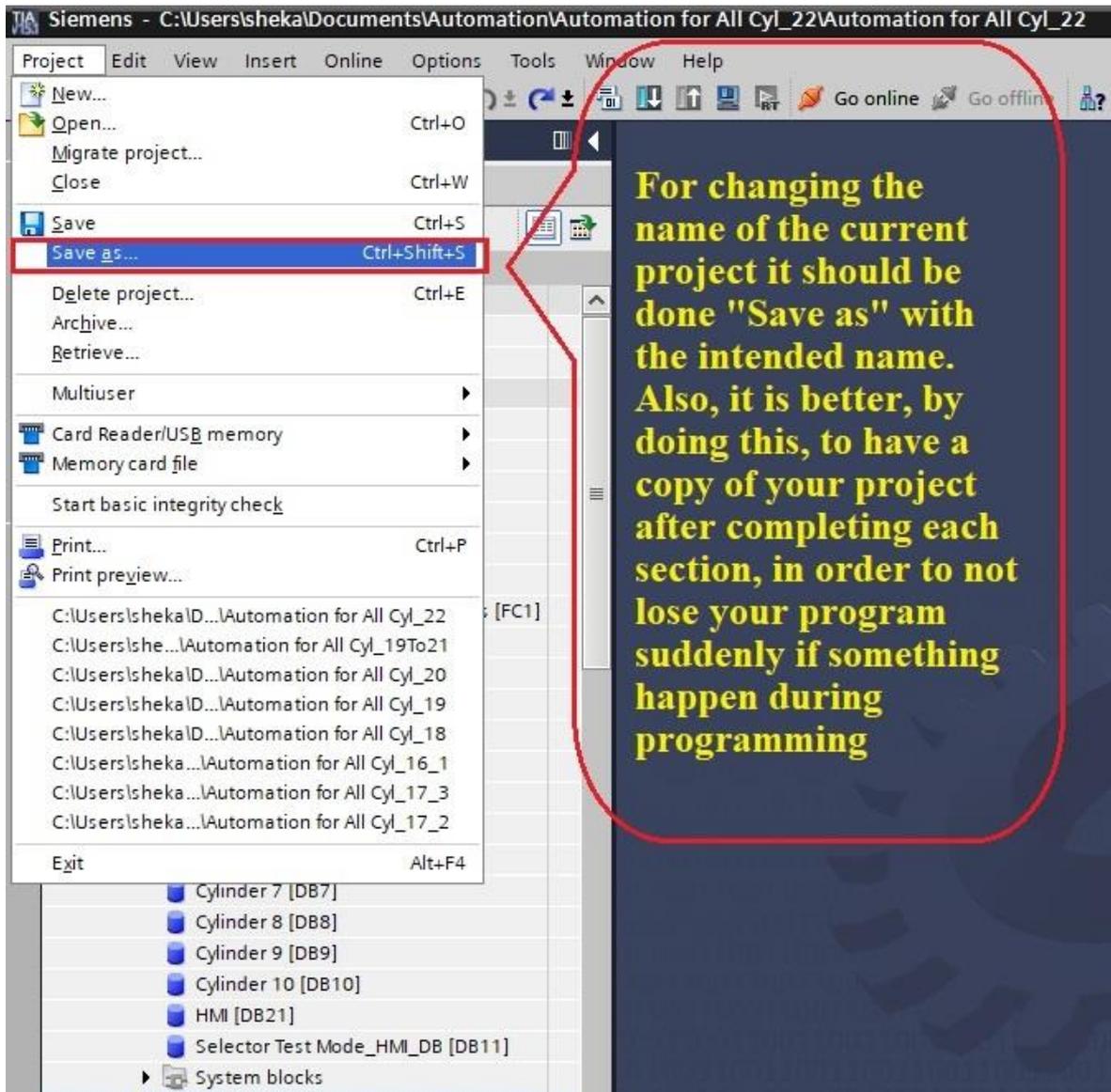
After installing and launching this software, choosing the second submenu in the START tab of the SW is called “Create new Project” by clicking on that you can open a new project. for creating the project, it is necessary to give it a name and the storage path to save the project that is usually pre-saved in the C drive, documentation directory. The first thing after creating the project is to configure and choose a PLC device /or device that has been provided. Then the second level is to write the PLC program. If there are some technology objects, in other words, motion control or something like this, it is necessary to be defined here at the 3rd level. And another important part of the project is designing and the configuration of an HMI, it should be done on the 4th level. In the last 5th level this SW, if there is any old project and you want to work on it you can click over this and open the existing project from your computer.

By clicking on device and network, you have three options the first one easy show all devices the second one is adding a new device and 3rd one easy configure networks. A help icon also is available to learn something new.

When we know the characteristic of the applied device we need to choose “Add new device” between all devices we can choose the one we need. While you open this menu, it is possible to choose three different kinds of devices. The first one is a PLC controller, the next one is the HMI display and the last one is PC systems. For each section there are many sub-devices that allow you to choose a specific device yet is needed in your project with different versions and order numbers. But you must be careful to choose the right one with the right firmware version. In this part, you will choose all needed devices that will be appeared in one of the submenus of the project which is called Device configuration on the left side of the software program under the menu of main PLC name.

While you have an idea of your program categorization on your mind it would be a good idea to start writing your program in some function blocks and you can call them in main OB1. It is possible to create a new block from one of the main folder menus which called “program blocks” after writing your program in the created block also it can be easily called into main OB1 by dragging and dropping or if it is necessary, you can also write your program directly in the main OB1. It is easy and possible to call any defined and needed variable from local or global variables from any database or PLC tag table or even HMI tag table.

While it is necessary to do changes to the existing project and having also the previous project as a base project, you can do the “Save As” on the project and save it with another name as it is shown in the figure below.



Figure

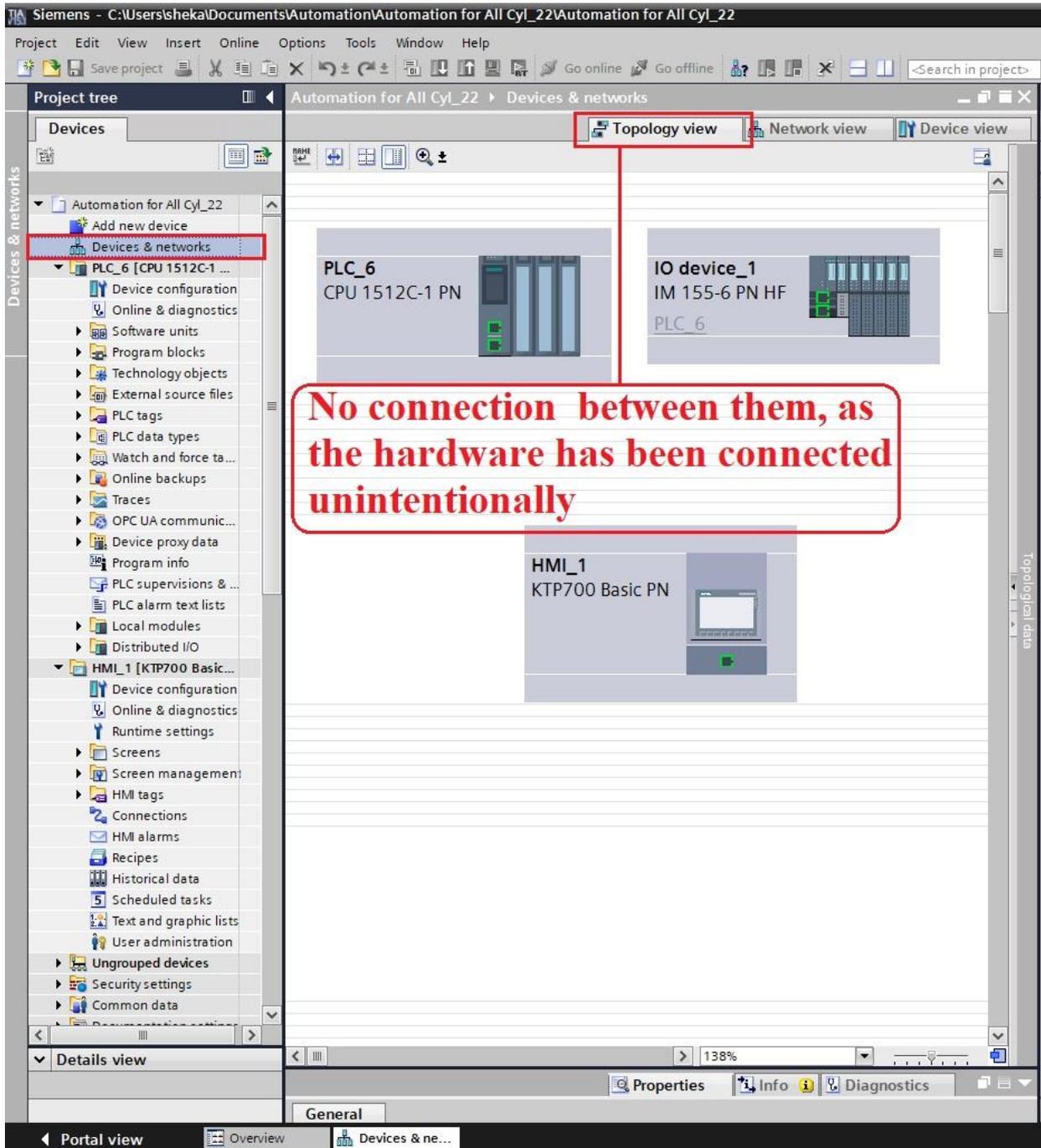
1-6-5- Device configuration, Interconnections and IP addresses

In the previous part, it is mentioned that for choosing a willing PLC device, you need to be in the Environment of “device and networks” which its main icon is located on the top left window.

While you are in this space, you can see and use three tabs on the top right. The first one is “Topology View” Which allows you to choose your devices and components from the hardware catalogue on the top right. By dragging and dropping them in the environment you can see them, and you can connect them together, also they will appear in the “Network view” and “Device view”.

One important point is after choosing your devices and bring them in the “Topology view” space, they must be connected exactly as same as the hardware component configuration. The port number for connecting devices together is the most important part which needs to be connected correctly with much attention. But if the needed components are called in the “Network View”, you can connect them as you wish, and the port number is NOT very important to contact the same as in the hardware device on the test bench. And in “Topology View” they should not be connecting for making a network, in reverse in Network View they must be connected. In the small projects also, it is possible to NOT bring the switch Scalance in the “Network review” space, however, there is a switch Scalance on the real test bench, it can be ignored.

It is very important to be noted that by connecting an Ethernet cable to the Switch Scale, and some small SW setting, I have controlled and programmed the test bench from a remote area.



Figure

While it is necessary to configure, modify or define some of the characteristics of each device, by clicking and choosing one of the devices then going to the “Device view”, You can see the single chosen device and all its properties, and some other information is available on the window at the button. There is another way to go through “Device view” directly, by doing double click on each device it goes directly to properties of each device and you can do the configuration as a previous way.

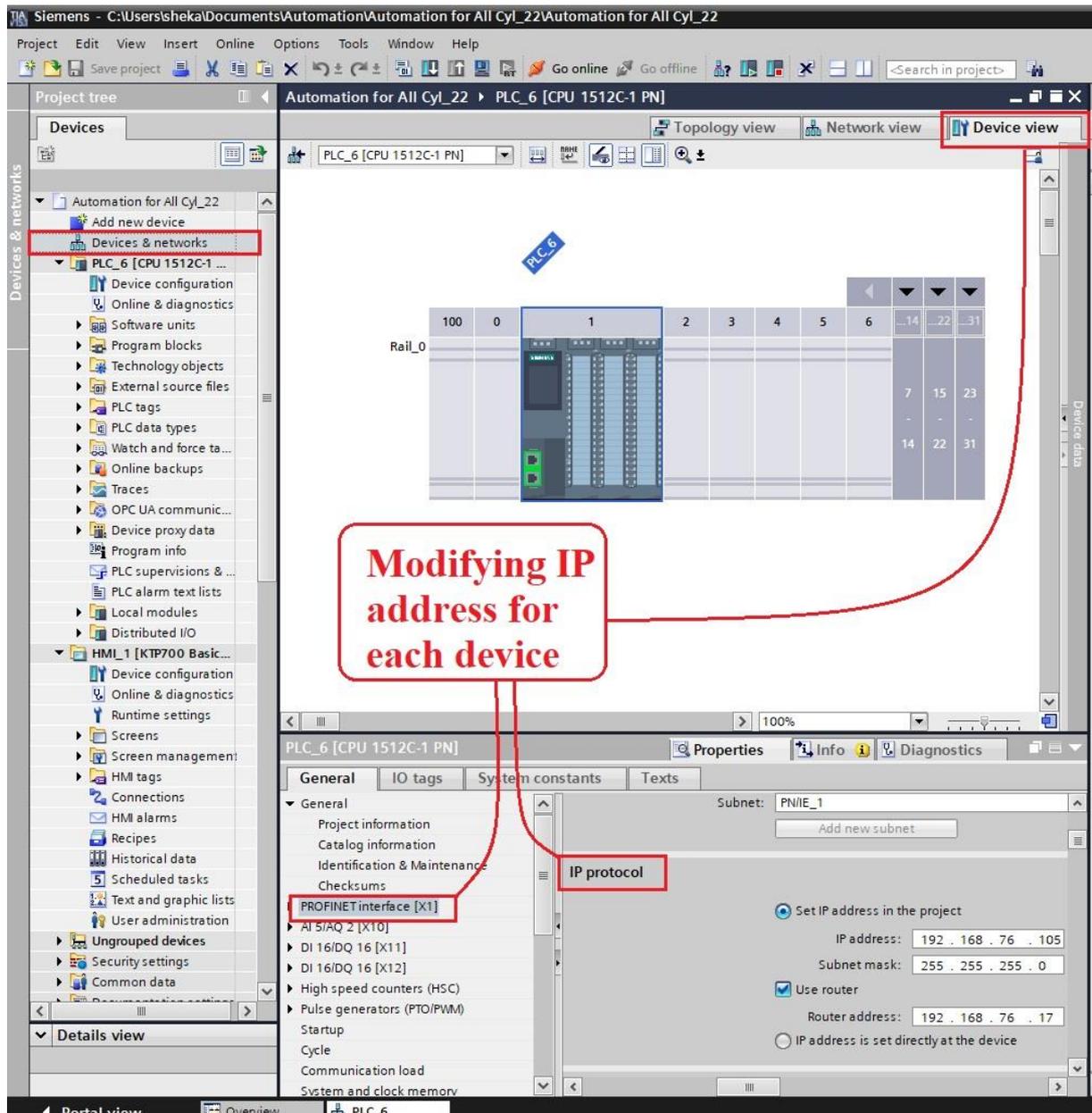
In this thesis, there are three devices that must be called in the “Network view” environment of the TIA portal SW. Although there is one Scalance switch that it would not be called here. As I have already explained that it is optional to have it in the software configuration while the interconnections have been done in the “Network view” instead of “Topology view”. In the “Topology view,” devices must NOT be connected to each other by green lines. Because the hardware configuration and connections not as same as software configuration.

In the Device View while the main PLC device has been chosen it is possible to see all its properties in the general tab and there are also other tabs on the top of the window below. In addition, for the configuration of each module, first, it should be selected the intended module and then all its characteristics will appear in the same window below it.

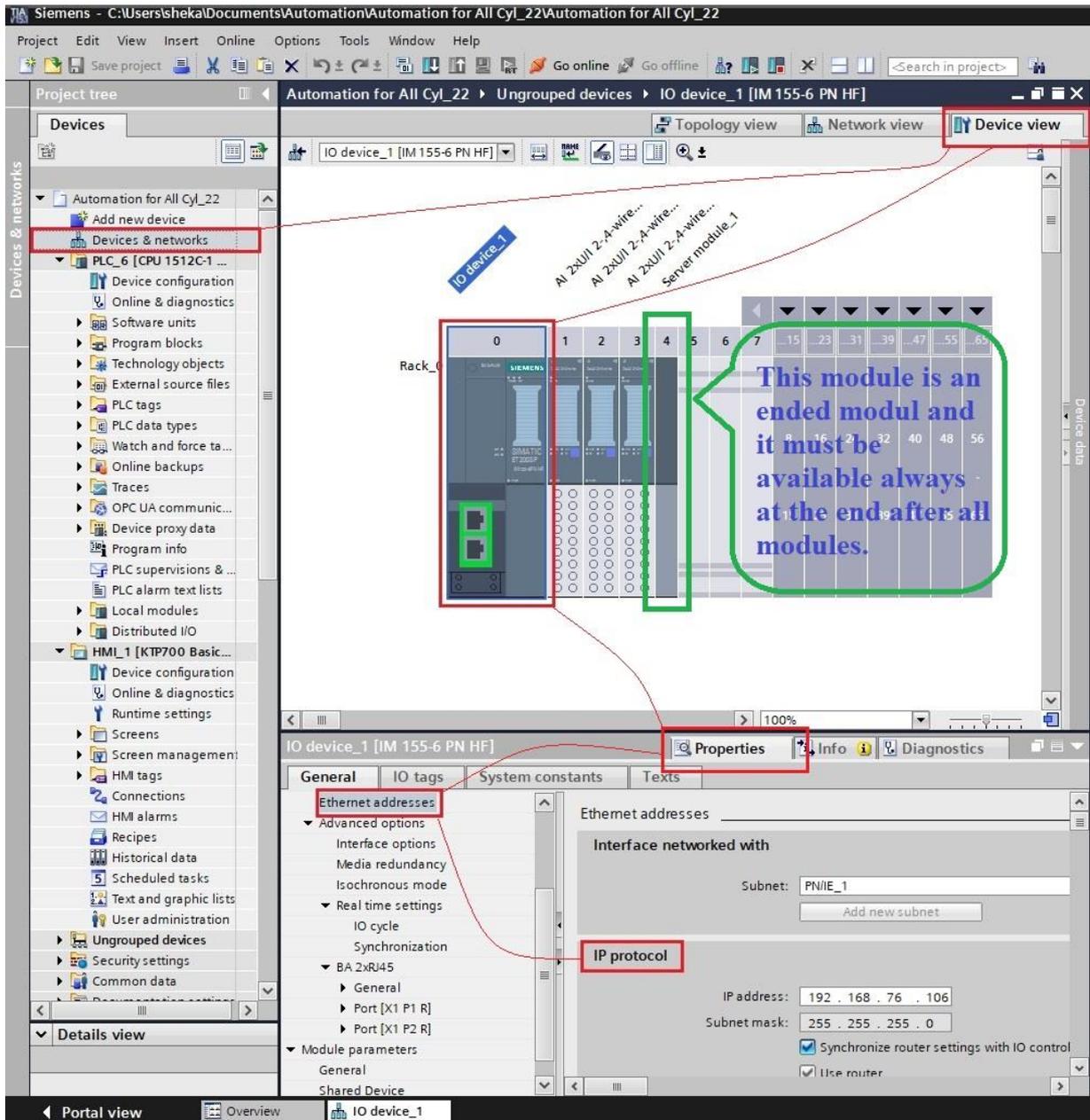
On any network, devices must be known by each other, due to this issue all devices must have their own IP addresses which can be fined by the programmer or it has a predefined IP address while it is calling.

In the case that the programmer should define it in the environment of “Device view” by Selecting the PLC device all its properties will appear in the window below it, by going to the properties tab and then general tab, there are many menus and sub-menus. The second menu and the general tab is related to PROFINET

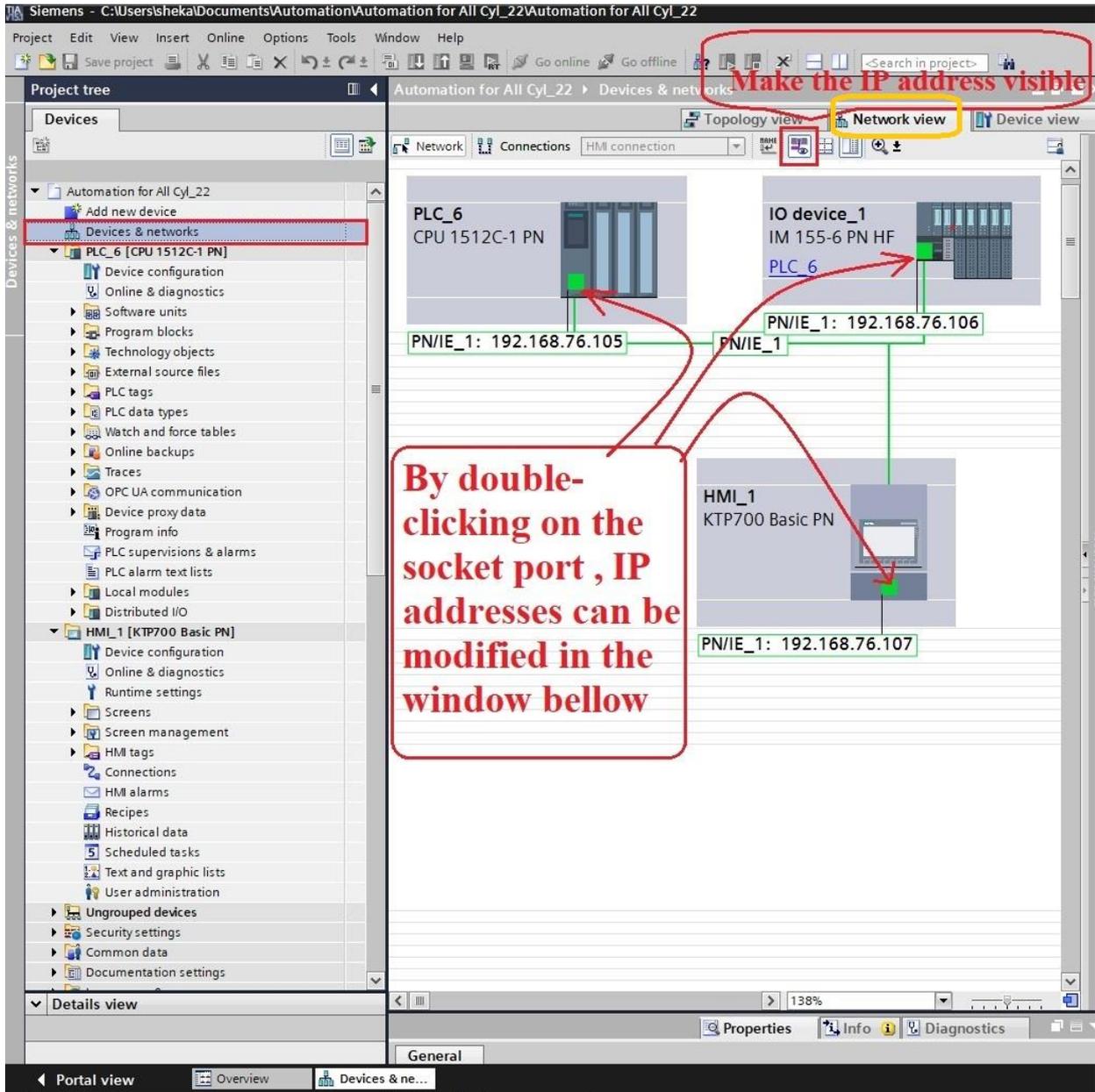
interface [X1]. Under this submenu, you can find IP protocol. in this part, IP addresses are shown, and it should be defined as you wish or as it is necessary to set and arrange with other devices on the whole network.



Figure



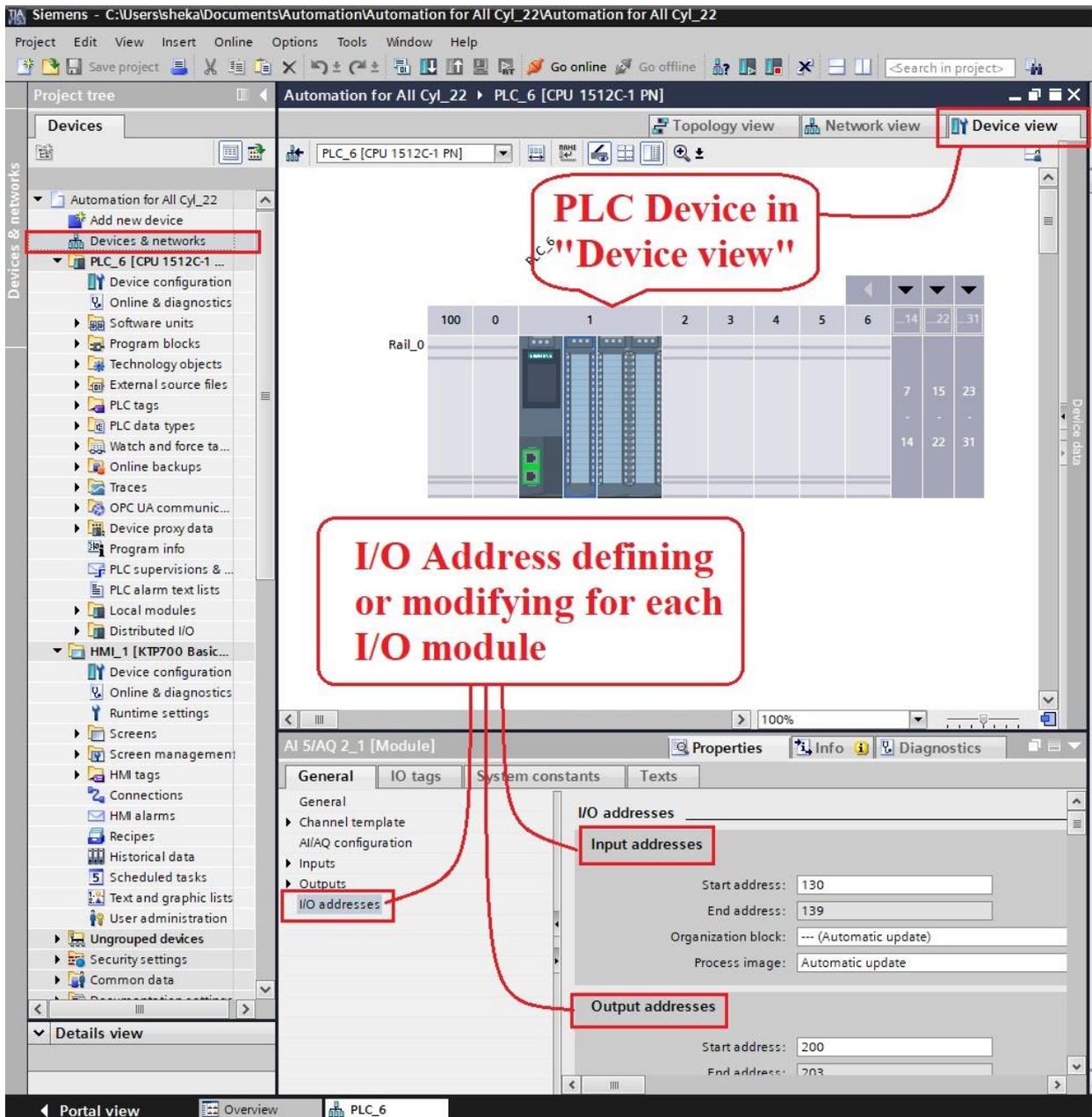
Figure



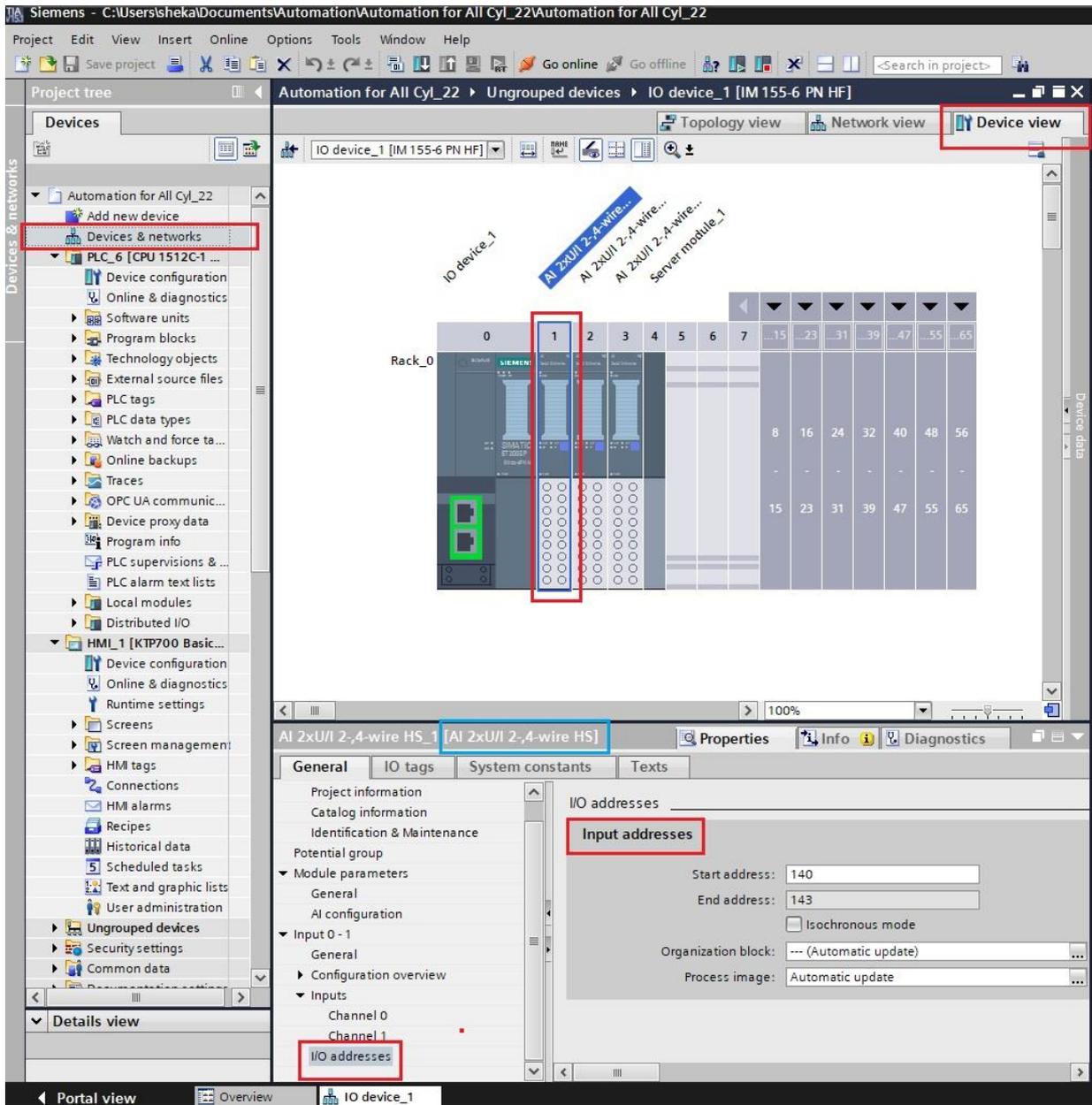
Figure

1-6-6- Memory configuration of PLC device

In all programmable devices, one of the significant parts of programming is memory definition and assigning. For memory definition and configuration of PLC S7-1500 while we are in the environment of the tab “Device view” under the “Device and networks” it allows defining then memory. After selecting the intended exact I/O module the sub-window below it from Properties tab --→ General tab ---→ The last subtab is called I/O addresses when it is clicked this Shown window indicates memory for input addresses and memory for output addresses. It is important to know both input and output memory can start from the same number bit because it has two different areas inside the PLC memory. for me digital input-output we must define one bit because they are Boolean, and every 8 bits make one byte. But for the analogue modules are totally different. In this case, for each analogue input two Byte, a Word data type has been assigned. For instance, if we assigned a Word data type number 130 for one analogue important it will occupy 2 bytes 130 and 132 but byte number 132 is not shown and the next one must be started from byte 133.



Figure



Figure

1-6-7- Functionality of blocks in the PLC program

In this section, it would be explained some basic starting ideas for programming.

The first one is a general idea about the program. Second one general idea to begin the programming. And the next one is function Blocks and Data Blocks.

1-6-7-1- General idea about the program

In the PLC program, there can be one main OB1 or more that all programs should be written inside them and other subroutines should be called inside them. In order to have an organized and categorized programmer, it is necessary to write the program in different networks and each network can consist of some Function Blocks, Functions, subroutines and programming lines. In addition, there would be created some data blocks which contain all the data for each block if it is necessary. It is possible to have one data block for many function blocks or having a separate data block for each function block. Creating a data block can be done automatically at the same time or manually later. Also, there are some different kind of data blocks with different instances as we need, we can choose one of them and create our willing data block. And in this case, when they want to create a data block that is related to inside the function blocks, we should create a multi-instance data block and if we want to use the data block in Main OB1, we should use a single instance data block.

1-6-7-2- How to begin programming & create OB, FB, FC or DB

For commencing to write the program 1st it is necessary to organize the programmer in different sections and the different networks then start creating function blocks and after creating a function blocks when we start writing the program in Main OB1 we can call them wherever we want and at that moment if it was an accessory to create data block, you can make it in the same time.

The main goal of creating the FB blocks or FC blocks is to categorize the program. While the program is in a different section it would be too easy to recognize any possible error or problem in the project. Besides, writing a program in different networks has the same purpose because when there is an issue in the project you can find the precise location in which Block and in which network of that block there is the error or mistake.

Here you can see how an FB or FC can be created in the project. After creating a new project, from the window on the left side of the TIA portal, by going into the folder of “program blocks” you should click on the sub-menu is called “Add new block”. Then a new small window would pop up which has 4 different types of Blocks. You can choose either OB, FC, FB or DB.

For instance, you choose the FB and click OK. Then an FB would be created, and it automatically would be open and is ready to start writing your program in different networks. To define any inputs, outputs, temps or any other local variable in different categorizations. Then you must go through the window which is located on the top of that block. All variables which are defined in the input section will have appeared on the input appearance part of the block and all variables which are defined in the output section will have appeared on the output appearance part of the block. Memory variables or any willing variables can be defined there and can be called inside the program however the local addresses can be directly called and used into the main OB. So, until now the FB is created in the project and it is appeared in the left window among sub-menus, under the menu of the folder “program blocks”. Now you can call it in the main OB or even in any other FB as sub-routine, by dragging and dropping it. It is noted that for any FB, automatically a DB block will be created with the same name. The main benefit of having FB is easy to use it many times in the program. Only after calling it, any time you should define variables on the inputs and outputs of any block by

addressing with a real variable address which should be the final control goal. It can be a real I/O address of a PLC device or as an HMI tag or etc.

1-6-7-3- Function Blocks and Data Blocks of the Project

This thesis contains one main OB1 block, one Function, two main Function Blocks, and 12 Data Blocks in the area of Program Blocks, one Function which is related to light mode under 3-position selector switch and its data block is located inside each Function Block it means it does not need separated data block. Two main Function Blocks Which one of them is related to the virtual 3-position selector switch on HMI and the next Function Block, which is very important and the main one, for each cylinder that should be copied and pasted, for all cylinders in different blocks in main OB1. Also, there are 12 Data Blocks in which 10 of them related to 10 cylinders one of them has been associated with HMI configuration and the next one is assigned to the action of virtual 3-position selector switch on HMI. Since now it has been explained only about the main things which must be done to organize categorize the Program Blocks area, however for creating a data blocker it is possible to create them when you're calling them in different networks at the same time. All these steps will be explained in a very detailed separately to show it is more understandable if someone wants to improve the test bench in upcoming days.

Siemens - C:\Users\sheka\Documents\Automation\Automation for All Cyl_22\Automation for All Cyl_22

Project Edit View Insert Online Options Tools Window Help

Save project [Icons] Go online Go offline [Icons] <Search in project>

Automation for All Cyl_22 > PLC_6 [CPU 1512C-1 PN] > Program blocks > Cylinder 1 [DB1]

Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

Cylinder 1 (snapshot created: 11/26/2019 3:12:52 PM)

	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Com
1	Input									
2	Leak Test	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
3	Wear Test	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
4	Both Test	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
5	START	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
6	Reset Cycle Total_Cabin	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
7	Reset Cycle Total_HMI	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
8	Door Limit Switch	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
9	EMERGENCY	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
10	Reset Counter Master_...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
11	Reset Count CYCLE_H...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
12	Ended Wear Cycles_IN	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
13	SO_PISTON IN	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
14	S1_PISTON OUT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
15	Reset Cycle Each Cylin...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
16	Reset Count Each Cyl...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
17	Setpoint Each Cycle	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
18	Setpoint Total Cycle	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
19	Pressure Manometer	Int	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
20	Output									
21	Wear Test LED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
22	Leak Test LED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
23	Both Test LED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
24	Finished Test LED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
25	Error LED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
26	HPV	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
27	LPV	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
28	ABV	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
29	PBV	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
30	Ended Wear Cycles_Out	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
31	DV	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
32	Time OUT_sec	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
33	Time IN_sec	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
34	Time OUT_min	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
35	Time IN_min	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
36	HMI_Cylinder Movem...	Int	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
37	Number of Each Cyl...	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
38	Number of Total Cycle...	Dint	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
39	Pressure Value_Input	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
40	Pressure Value_Anteri...	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			

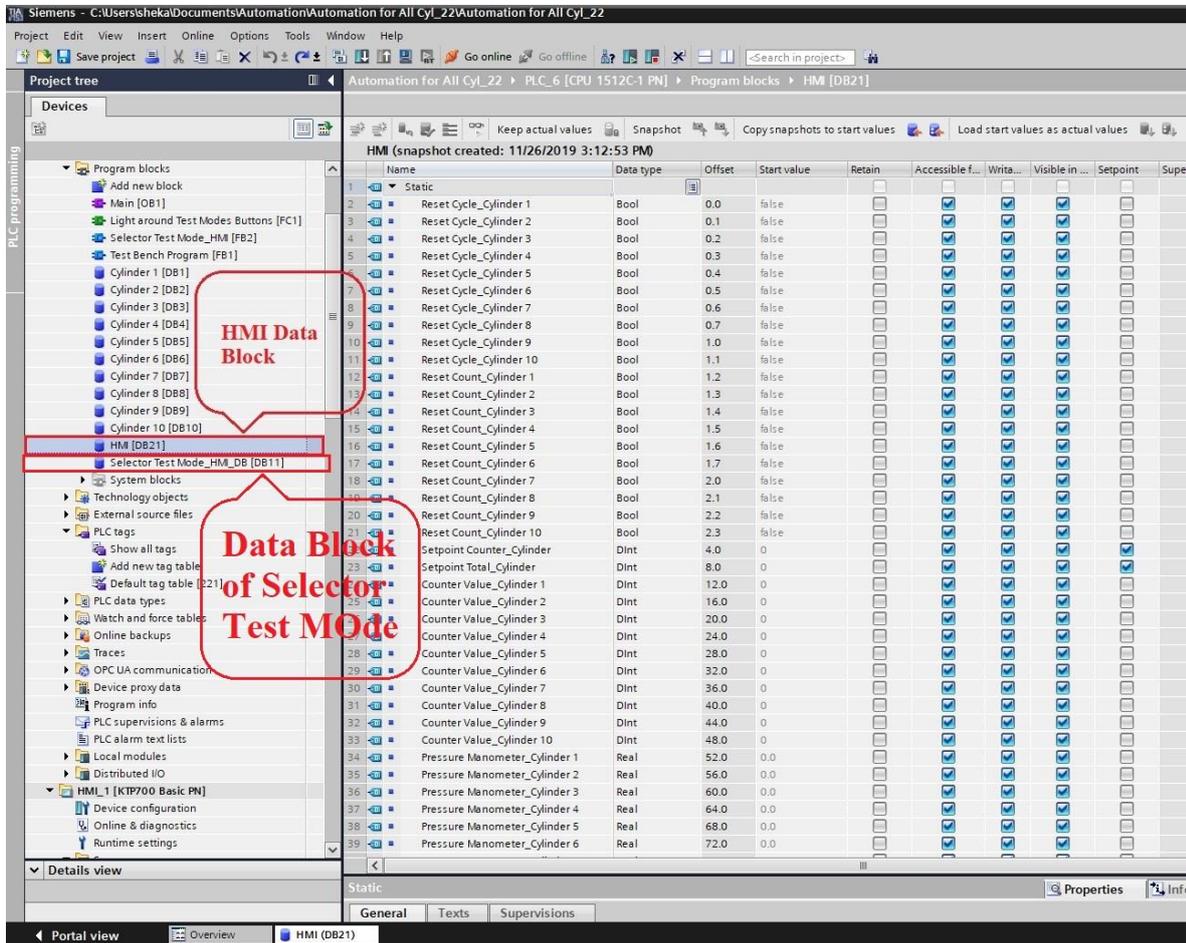
Input Properties

General Texts Supervisions

Portal view Overview Cylinder 1 (D...

Data block for each cylinder & for all, internal variable are the same

Figure



Figure

A. 1-7- Functionality of whole program Blocks and Networks

In this program, there are many networks in each block and there are also many blocks in Main OB1, next to several Ladder network programs.

1-7-1-PLC Tag table

In any controller program, many variables must be defined that some variables are global, and some other variables are local. Local variables must be defined inside each block which will be located inside the data block of the same function block. This is the reason they are called local variables.

But PLC Tag table allows predefining all intended global variables of the program which can be called in all blocks and inside Main OB1. In other words, when all global variables are defined in PLC tag table, it makes easy the programming because when programmer wants to assign any analogue or digital and input or output variable, it is not necessary to memorize the exact memory address for all variables, so it is possible to call any variable by the name. One of the most significant benefits of creating the tag table in advance is that while it will be needed to change the memory address of any variable, it is possible to do any changes whenever it is a necessity and easily. Also, it might happen in reverse, it means, if the dedicated names need to be changed in any reason, easily they can be changed in the PLC tag and there would not happen any problem and/or any changes inside the program, and it itself will be automatically replaced with the previous name and with the same memory address.

Siemens - C:\Users\sheka\Documents\Automation\Automation for All Cyl_22\Automation for All Cyl_22

Project Edit View Insert Online Options Tools Window Help

Save project Go online Go offline <Search in project>

Automation for All Cyl_22 > PLC_6 [CPU 1512C-1 PN] > PLC tags

Tags User constants System constants

PLC Tag Table

Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Super...
1 Select leak Test	le Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2 Select Wear Test	le Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3 Start Button	le Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4 Reset Cycle Button	le Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5 S0-Cylinder 01	le Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6 S1-Cylinder 01	le Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7 S0-Cylinder 02	le Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8 S1-Cylinder 02	le Bool	%I1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9 S0-Cylinder 03	le Bool	%I1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10 S1-Cylinder 03	le Bool	%I1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11 S0-Cylinder 04	le Bool	%I1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12 S1-Cylinder 04	le Bool	%I1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13 S0-Cylinder 05	le Bool	%I1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14 S1-Cylinder 05	le Bool	%I1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15 S0-Cylinder 06	le Bool	%I1.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16 S1-Cylinder 06	le Bool	%I2.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17 S0-Cylinder 07	le Bool	%I2.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18 S1-Cylinder 07	le Bool	%I2.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19 S0-Cylinder 08	le Bool	%I2.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20 S1-Cylinder 08	le Bool	%I2.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21 S0-Cylinder 09	le Bool	%I2.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22 S1-Cylinder 09	le Bool	%I2.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23 S0-Cylinder 10	le Bool	%I2.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
24 S1-Cylinder 10	le Bool	%I3.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
25 Door Limit Switch	le Bool	%I3.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26 Emergency	le Bool	%I3.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
27 Pressure Manometer 01	le Int	%IW130		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
28 Pressure Manometer 02	le Int	%IW132		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
29 Pressure Manometer 03	le Int	%IW134		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
30 Pressure Manometer 04	le Int	%IW136		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
31 Pressure Manometer 05	le Int	%IW138		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
32 Pressure Manometer 06	le Int	%IW140		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
33 Pressure Manometer 07	le Int	%IW142		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
34 Pressure Manometer 08	le Int	%IW144		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
35 Pressure Manometer 09	le Int	%IW146		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
36 Pressure Manometer 10	le Int	%IW148		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
37 Wear Test LED	le Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
38 Leak Test LED	le Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
39 Both Test LED	le Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
40 Finished Test LED	le Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

By clicking on each tab, variables will be ordered by that column

Select leak Test [PLC tag]

Properties Info Diagnostics

General Texts Supervisions

Portal view Overview PLC tags

Figure

1-7-2-Main [OB1]

This part of the software is the main environment to write down, design and/or call any function block to be executed. For this program, in Main OB space any command will be executed from the left to the right in each line and from up to down in each column. The visible icon colour of this submenu is purple. That is located under the “Program Block” folder in the top of the left window under the “PLC Project” main folder.

Regarding this thesis, there are many blocks in main OB1 which some of them are related to each cylinder exactly and some of them are common between all cylinders, the total number of cylinders is 10 or having one common function for all cylinders.

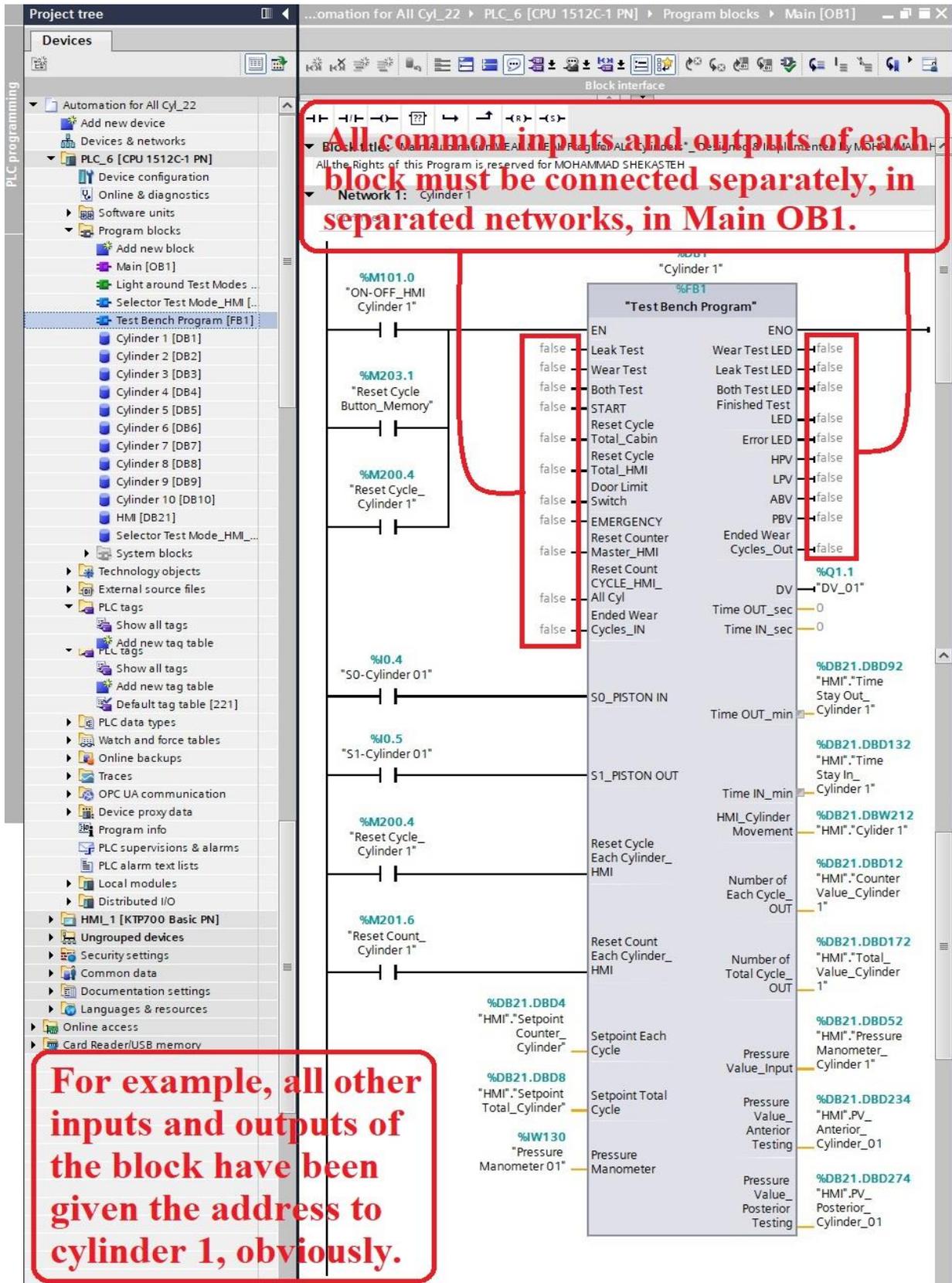
In the Main OB1 environment of this program, there are 34 networks totally, which the first 10 blocks exactly and directly related to each cylinder, and who 24 other networks are common for all cylinders. In this part I will try to illustrate and give the explanation other all networks’ contents, one by one, to clarify the exact direction of the program.

1-7-2-1- Content of “Main OB1” totally 34 networks

Network 1 to 10;

As has been mentioned before, in each of these networks there is one block and, all 10 blocks, in all 10 networks are the same as each other. It means all local variables and local programming of all 10 blocks are the same. The only difference is to connect the input and output of each block to the exact real physical and global address which must have to be predefined in the global “PLC Tags” Table variables. In the figure below it is shown the common Block with associated addresses for cylinder number one, as an instance.

All common inputs and outputs of each block must be connected separately in separate networks in Main OB1. For example, in the figure below, all uncommon inputs and outputs of the block have been given the address to cylinder 1, it is obvious. Some of the uncommon I/Os have called from the HMI tag table, some of the other I/O is called from the intended Data Block which is local variables. But usually, they are called from PLC Tag Table that has defined in advance.



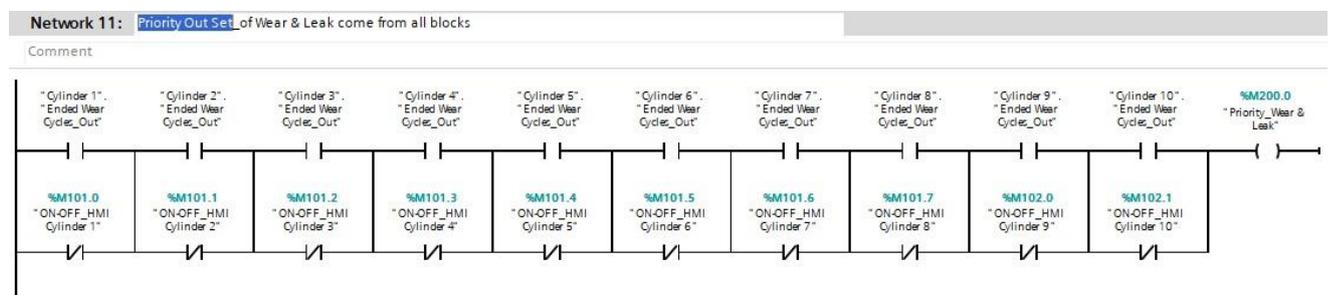
Figure

Network 11; Set Priority conditions in “Both Modes” test

Only for the “Both Modes” test, there must be the availability of the priority for all cylinders while they are going to swap the test from WEAR test LEAK test. In other words, in this mode only when all cylinders have finished the WEAR test then they can be allowed to change from one test mode to another test mode.

In this case, I have designed, inside all blocks, in the way that there has been set the memory for the WEAR test and send the signal in the output section of each block called “Ended Wear Cycle_Out”, if WEAR test finishes, in each program block. So, I have put all those memories, which are local variables in different data blocks of cylinders, in series that when all the cylinders would finish the first test mode, they will close a path to set another global memory. This memory will send the signals to one of the inputs of all blocks that called “Ended Wear Cycle_IN”, which is related to start the next mode test, it will be explained in the next network (12).

In addition, there is another significant point that Has been considered in this network. there are 10 toggle switches (On/Off), on HMI, which disable and the deactivate only each block, if there is any problem for each of cylinders or if it is necessary to test some of them. Hence, I have put the memory contact of each toggle switch in parallel with the memory contact of the same cylinder contact of “Ended Wear Cycle_Out”. It allows while the cylinder turns off, this contact acts and closes the path as an “Ended Wear Cycle_Out” contact memory.



Network 12; Finished Priority signal to start “LEAK Mode”

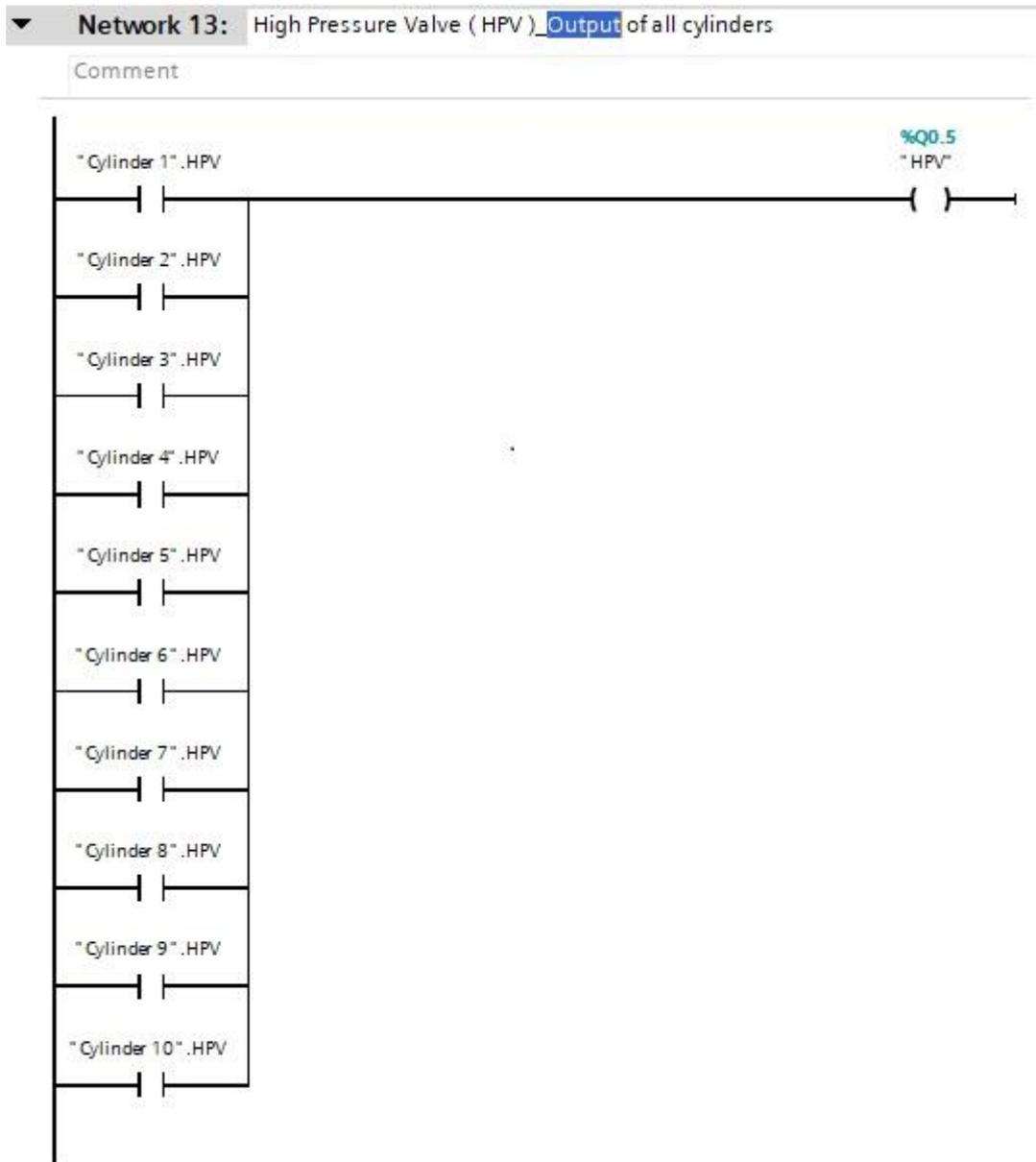
As was mentioned in the previous part, this is a network that after receiving the priority signals of all cylinders in the WEAR test, a global memory had been activated in previous networks. And here, an open contact of that memory is applied to send the signals to one of the inputs, which is called “Ended Wear Cycle_IN”, of all blocks exactly and precisely in the same moment, for commencing the second test mode “Leak”. In order to do this, it needs to make a network connecting one single input condition from the defined global memory in the previous network to connect to 10 parallel outputs which are called from each cylinder block. Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely.

By executing the programme of this network all cylinders start doing the 2nd test at the same time. It means, two supplying valves, that one of them is a High-pressure valve and another one is a Low-pressure valve, they will not be activated at the same time, as they are common valves for all cylinders.

Network 13: Activating output HPV

There is a difference between inputs that need to be paralleled and the output which needs to be parallel, for making a network to send or receive the signal at the same moment.

This network is related to activate the single High-Pressure Valve. In order to do this, it is necessary to create 10 parallel inputs normally open contacts, which come from each one of the outputs of all cylinders’ blocks and then connect it to one single output that is one global memory and has been already defined in PLC Tag Table.



Network 14: Activating output LPV

This network is related to energize the single Low-Pressure Valve. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinder blocks and then connect it to one single output that is one global memory and has been already defined in PLC Tag Table.

Network 15: Activating output ABV

This network is related to energize the single Anterior Block Valve. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders' blocks and then connect it to one single output that is one global memory and has been already defined in PLC Tag Table.

Network 16: Activating output PBV

This network is related to energize the single Posterior Block Valve. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders' blocks and then connect it to one real single physical address output that is one global memory and has been already defined in PLC Tag Table.

Network 17: Activating inputs "Wear select Mode"

The functionality of this network is to send a real single input signal to the inputs "WEAR test Mode" test of all blocks. Hence, after putting some memory conditions at the beginning of the network on the path, we should put the real physical input address which is a global memory and has been defined in the PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 18: Activating inputs "Leak select Mode"

This network is related to transfer a real single input signal to the inputs "LEAK select Mode" of all blocks. Hence, after putting some memory conditions at the

beginning of the network on the path, we should put the real physical input address which is a global memory and has been defined in the PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from the input port of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 19: Activating inputs "Both select Mode"

The functionality of this network is to send a real single input signal to the inputs "BOTH test Mode" of all blocks. Hence, after putting some memory conditions at the beginning of the network path, we should put the absence conditions of real physical inputs address which is the absence of two other test modes. Then those should be connected to 10 parallel outputs which are called from the input port of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just next to the input of the blocks, some of the blocks would be able to transfer the data to the actuator for energizing them and some others are not able to be executed at the willing time.

Network 20: Activating inputs "Door Limit switch"

This network is related to transfer a real single input signal to the inputs "***Door Limit switch***" of all blocks. This is important due to, it is related to human safety for all cylinders. By opening the guard window in front of all cylinders, it will stop the whole cylinders at the same moment. Hence, after putting the memory condition at the beginning of the network on the path, this memory is receiving a

command from the door limit switch, we should put the real physical input address which is a global memory and has been defined in PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from the input port of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 21: Activating inputs "Emergency Button"

This network is related to transfer a real single input signal to the inputs "***Emergency Button***" of all blocks. This is important due to, it is related to human safety and security for all cylinders during the test. By pushing the Emergency button on the cabinet switchgear, it will stop the whole cylinders at the same moment. Hence, after putting the memory condition at the beginning of the network on the path, this memory is receiving a command from the Emergency button, we should put the real physical input address which is a global memory and has been defined in PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from the input section of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 22: Activating inputs "Reset Counter Master_HMI"

This network is related to convey a single input signal from HMI panel to the inputs "***Reset Counter Master_HMI***" of all blocks. By pushing the Reset Counter Master_HMI on the HMI panel, it will reset and send a zero to the whole

cycle and total counters at the same moment. Hence, after putting the memory condition at the beginning of the network on the path, we should put the real input address memory which is a global memory and has been defined in the PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from the input section of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 23: Activating inputs "Reset Counter Each Cycle_HMI"

This network is related to convey a single input signal from HMI panel to the inputs “**Reset Counter Each Cycle_HMI**” of all blocks. By pushing the Reset Counter Each Cycle_HMI on the HMI panel, it will reset and send a zero to the whole **only** cycle counters, but not to the total counters, at the same moment. Hence, after putting the memory condition at the beginning of the network on the path, we should put the real input address memory which is a global memory and has been defined in the PLC Tag Table. Then those should be connected to 10 parallel outputs which are called from the input section of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 24: Activating inputs "Reset Cycle Total_HMI"

This network has a key role of functionality to send a single input signal from HMI panel to the inputs “**Reset Cycle Total_HMI**” of all blocks. By pushing the Reset Cycle Total_HMI on the HMI panel, it will reset the whole cycles of the

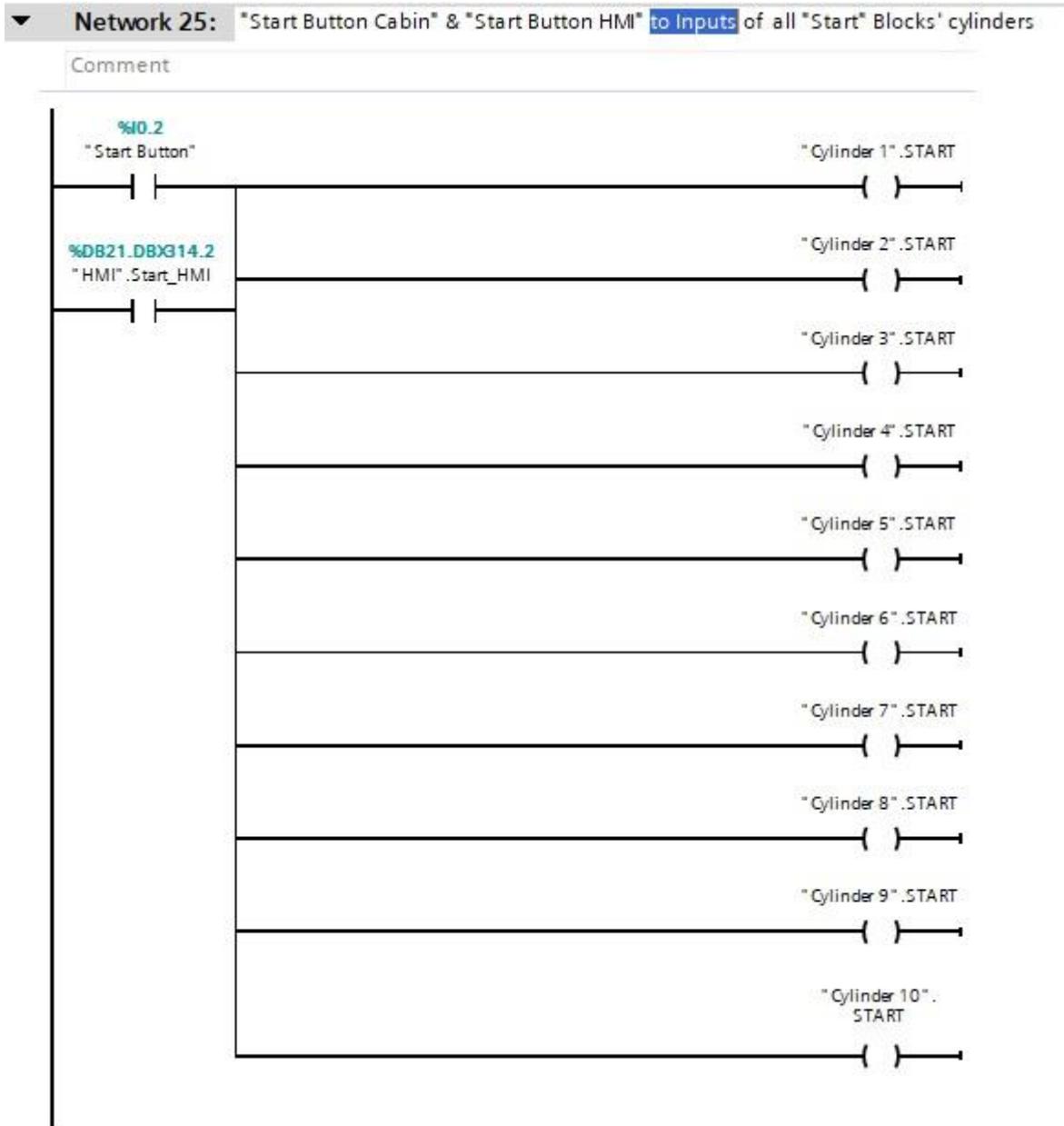
whole program in any mode at the same moment of pushing. Hence, after putting the reset memory condition at the beginning of the network on the path, that we should put the real physical input address which is a global input memory and has been defined in PLC Tag Table. Then that should be connected to 10 parallel outputs which are called from the input section of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

Network 25: Activating inputs "Start Button"

This network also is playing a significant role in functionality to send a single input signal from HMI panel and physical cabinet panel to the inputs “**Start Button**” of all blocks. By pushing the “Start Button” on the HMI panel or physical cabinet panel, it will run the selected cycles of the whole program in any mode at the same moment of pushing. Even if there is a stopped cylinder that has stopped by any sensor problem, among all other working cylinders, after troubleshooting sensors, by pushing the start button while other cylinders are working, the stopped cylinder would be run next to others. Hence, after putting the start memory conditions in parallel at the beginning of the network on the path, which is the real physical input address and is a global input memory that has been defined in PLC Tag Table. Then that should be connected to 10 parallel outputs which are called from the input port of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.



Network 26: Activating inputs "Reset Cycle Total_Cabinet"

This network has a key role of functionality to send a single input signal from HMI panel to the inputs "Reset Cycle Total_Cabinet" of all blocks. By pushing the Reset Cycle Total_Cabinet on the electric physical switchgear, it will reset the whole cycles of the whole program in any mode at the same moment of pushing. Hence, after putting the reset memory condition at the beginning of the network on the path which is the real physical input address and is a global input memory

that has been defined in the PLC Tag Table. Then that should be connected to 10 parallel outputs which are called from the input section of each cylinder block.

Only by this method signals can be transferred to the inputs of all blocks correctly and at the same time precisely. Otherwise, if we associated any memory address exactly just to the input of the blocks, some of the blocks might be executed and some others are not able to be executed.

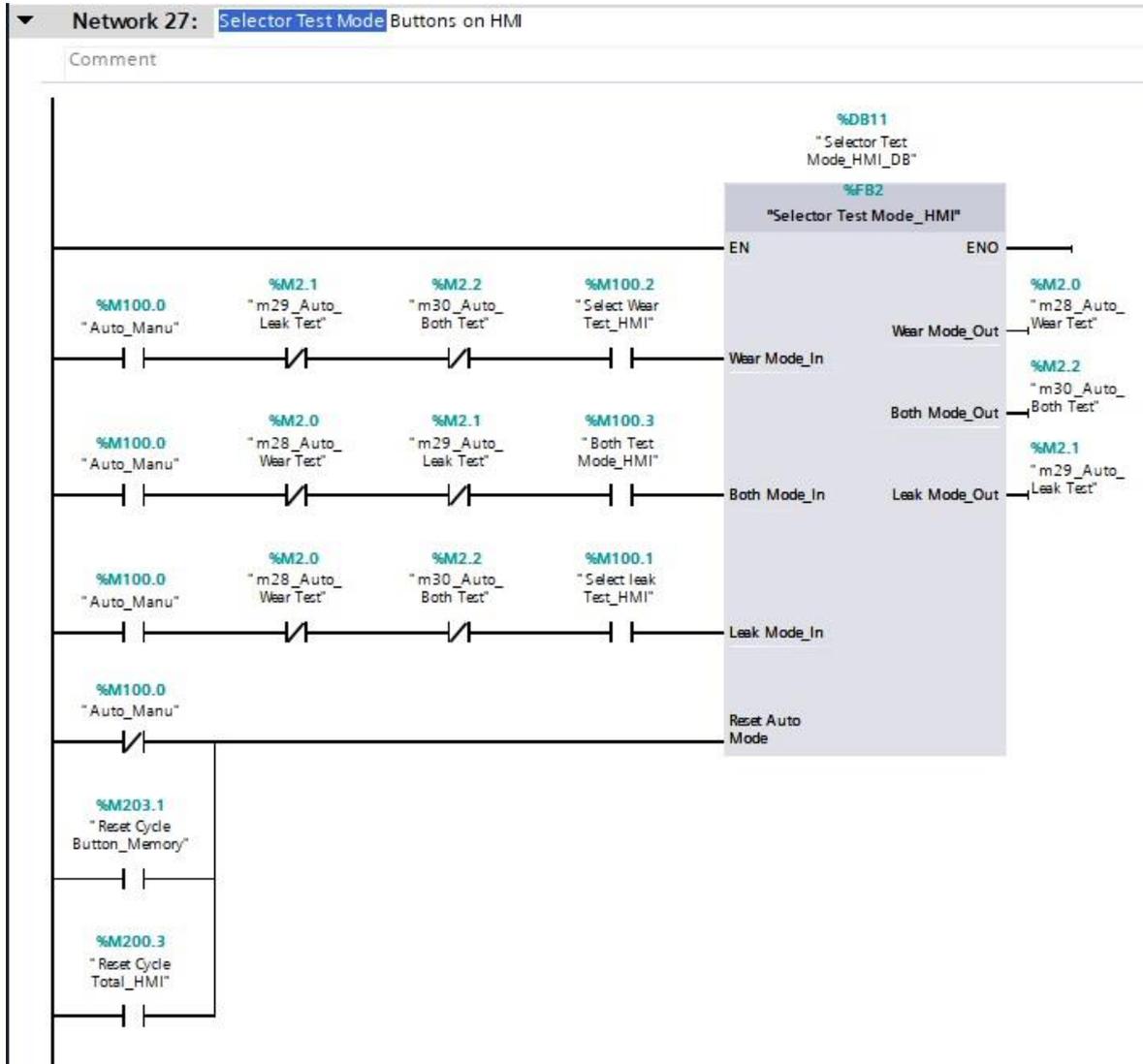
Network 27: Program block of “Selector Test Mode_HMI”

The significant role of this block is to control and swapping the test mode from HMI and control remotely from personal PC from anywhere around the world, however, all the states of the test bench can be seen too, while there is access to the internet connection. It is noted that This block can be executed **only** when Auto_Manual toggle switch on the HMI, is on the **Auto** mode. Otherwise, if it is on the manual mode, for changing the test mode it is necessary to swap the mod only from physical cabinet panel, however, it is necessary to reset the previous test mode in advance.

This block consists of five inputs and four outputs. The design of this program is in the way that three modes as three inputs can be activated by some conditions on their paths. The first input enables the block for executing the internal functions. The last input is the reset mode only for this block that will be received the command from three parallel contacts, by changing 1st one Auto_Manual memory or 2nd one Reset Cycle_Cabin memory or 3rd on Reset Cycle_HMI memory, the test mode and this block will be reset. As any reason for changing the test mode in any case first, it must be reset.

There are four outputs for this block. The first output is only a memory to show the block is active. And the three other outputs which exactly related to three

modes. The open contact of each virtual output mode must be paralleled with the exactly three physical input contacts in the program block.



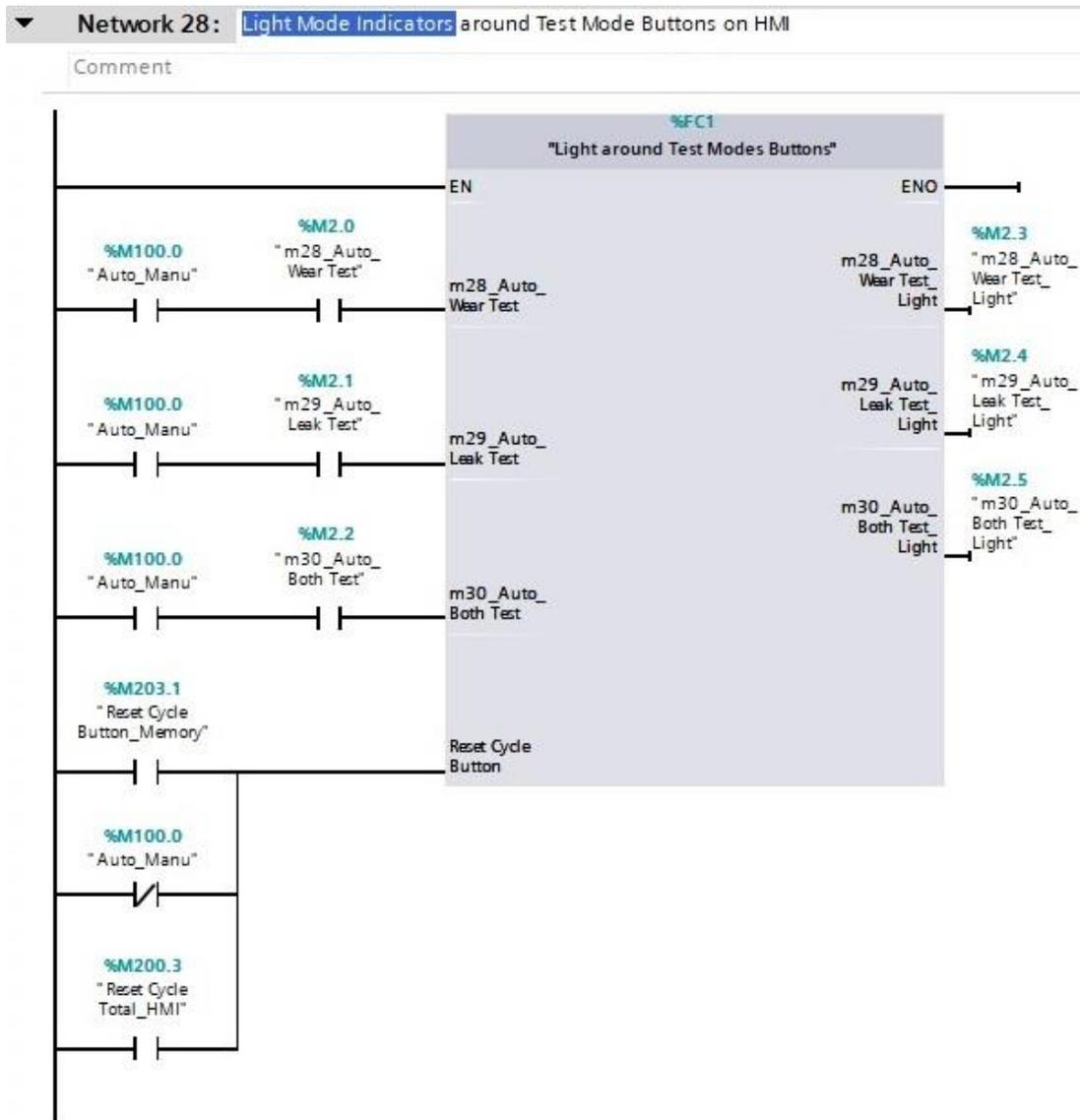
Network 28: Program block of “Indicator Light Mode_HMI”

It is essential to indicate; in which mode the test bench is working at the same moment. In order to illustrate this issue, a function block has been designed in this network that clarifies the mode of the test bench.

This block is somehow like a previous block and network with a little change.

There are five inputs and four outputs. The first important and then enable the functionality of the block. And three of the other inputs are sending the data from the previous network to activate a memory for each mode light, however, this block only can be executed in Auto mode. The last input is the reset mode only for this block which will be received the command from three parallel contacts, and by changing 1st one Reset Cycle_Cabin memory or 2nd one Auto_Manual memory or 3rd on Reset Cycle_HMI memory, the test mode and this block will be reset.

There are four outputs for this block. The first one is a memory to show the block is active. And the three other outputs which exactly related to three Light Modes on HMI to illustrate the test mode. The contact of each virtual output mode must be attached to each Indicator Light ribbon around test mode buttons on the HMI screen.



Network 29: Activating output “Wear Test LED”

This network is related to energize the Wear Test LED in the physical output of PLC and an LED indicator on the HMI screen. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders’ blocks and then connect it to one real single physical output and another one on signal light on HMI screen, that first one is one global memory and has been already defined in PLC Tag Table and the second one with having one more condition is attached to HMI Light LED.

Network 30: Activating output “Leak Test LED”

This network is related to energize the Leak Test LED in the physical output of PLC and an LED indicator on the HMI screen. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders’ blocks and then connect it to one real single physical output and another one on signal light on HMI screen, that first one is one global memory and has been already defined in PLC Tag Table and the second one with having one more condition is attached to HMI Light LED.

Network 31: Activating output “Both Test LED”

This network is related to energize Both Test LED in the physical output of PLC and an LED indicator on the HMI screen. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders’ blocks and then connect it to one real single physical output and another one on signal light on HMI screen, that first one is one global memory and has been already defined in PLC Tag Table and the second one with having one more condition is attached to HMI Light LED.

Network 32: Activating output “Finished Test LED”

This network is totally different because it is only an idea to set the final Finished test LED on the cabinet panel. While all cylinders start working at the same time but as the velocity of each cylinder is different, they would finish the test at a different time. in order to energize the Finished LED Test after finishing all cylinders, all Finished Test LED outputs of all cylinders’ blocks should be put in series together. So, in this way when all blocks have been executed the real physical Finished Test LED will be energized and remain energized until the Reset button is pushed, or another test mode is run.

Network 33: Activating output “Error Test LED”

This network is related to energize the Error Test LED in the physical output of PLC. In order to perform this, it is necessary to call 10 parallel input normally open contacts, which come from each one of the outputs of all cylinders’ blocks and then connect it to one real single physical output that is a global memory and has been already defined in PLC Tag Table.

Network 34: Activating one memory for “Reset Button”

The goal of this network is to convert the normally closed contact of the reset cycle button into a normally open contact to prevent doing mistakes. And in the whole program, this memory is used instead of the Reset pushbutton memory address.

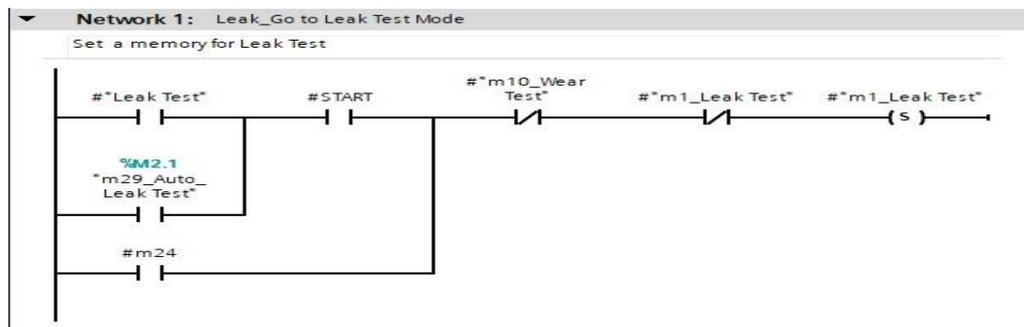
1-7-3-Test Bench Program Block _Function Block [FB1]

In each programme, there are some function blocks and some subroutines. In this program, the “Test Bench Program Block” is the main block for each cylinder that will be used for all 10 cylinders. In the upcoming paragraphs, all the explanations of all networks in this block would be available.

This function block consists of five main sections and it totally 45 networks. In the 1st 14 networks “LEAK Test Mode” programming will be explained, in the next 10 networks “WEAR Test Mode” will be illustrated and the 3rd part “Both Test Mode” will be described in the next 8 networks. All output LED indicators will appear in the next 6 networks and finally the last section regarding Scaling analogue input for each cylinder or block plus designing a program for movement of the cylinder on the HMI screen also the last network is relating to the minimum pressure limitation.

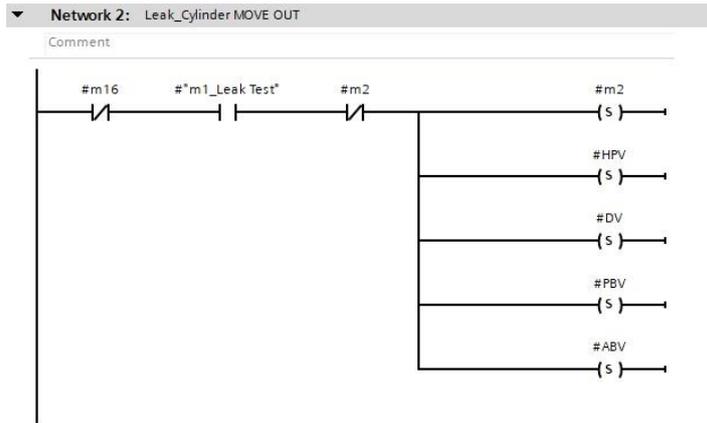
1-7-3-1- Network 1 to 14 is about “LEAK Test Mode”

N-1- The goal of the first network is to set the memory for starting the Leak mode. In the beginning, 3 parallel open contacts seen, first one is input that comes from the selector mode from the physical selector mode on electrical cabinet panel, second one is coming from virtual selector mode on HMI screen panel, and the 3rd one is coming from both modes test which tends to run this mode automatically and successively with another test together. there are 2 close contacts that are series Before the output which is related to prevent on programs mode not interrupting each other and not be started again. Also, it is clear the start contact is located after parallel contacts in the beginning.

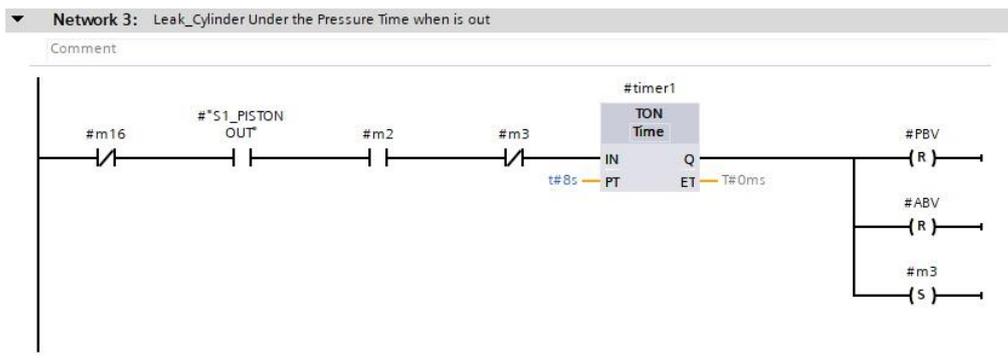


N-2- In this network, all valves which are related to moving OUT the piston must be ON, but some conditions on their path are necessary for which one of them is related to the previous network and one other is related to memory in this network that prevents receiving a signal again.

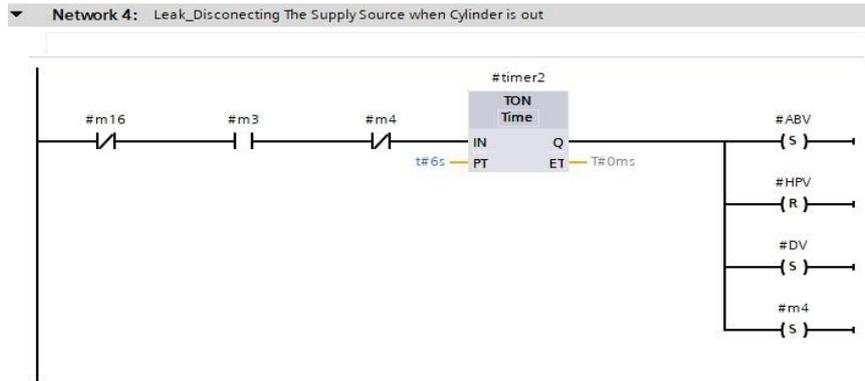
The first close contact condition is to disconnect the whole circuit with no networks of this test mode. So, at the beginning of all networks in this mode, it has been located to control the networks and the mode.



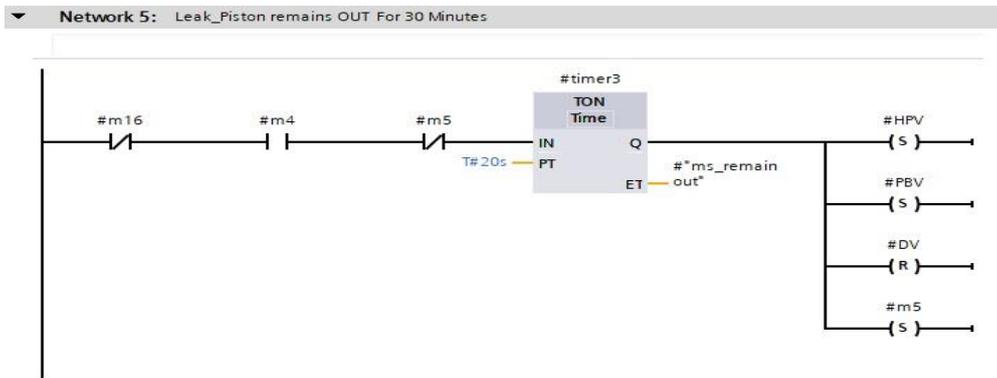
N-3- After pushing the cylinder out, it must be under high pressure for some seconds, so this time is doing the right action and by finishing the time, 2 blocking valves are reset and run the next step.



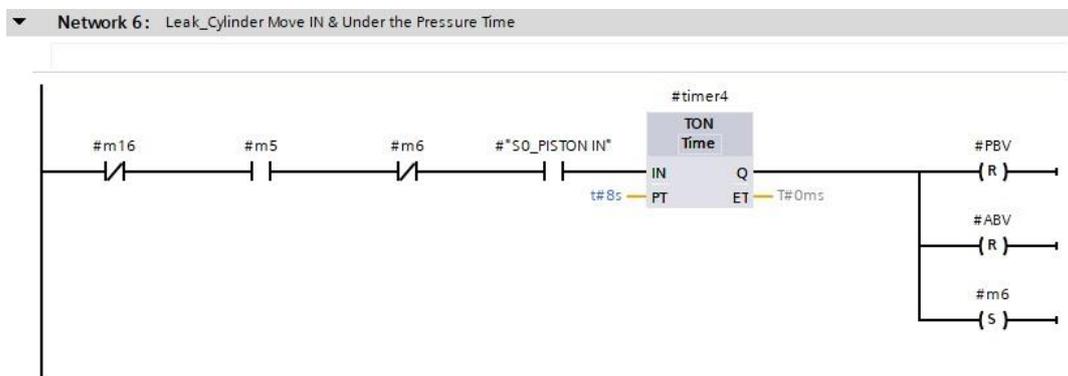
N-4- The timer in this network disconnects the main air supply by resetting a high-pressure valve and it opens the path to the front chamber by activating anterior block valve. Also, there is a memory to not allow starting it again and then activating the next network.



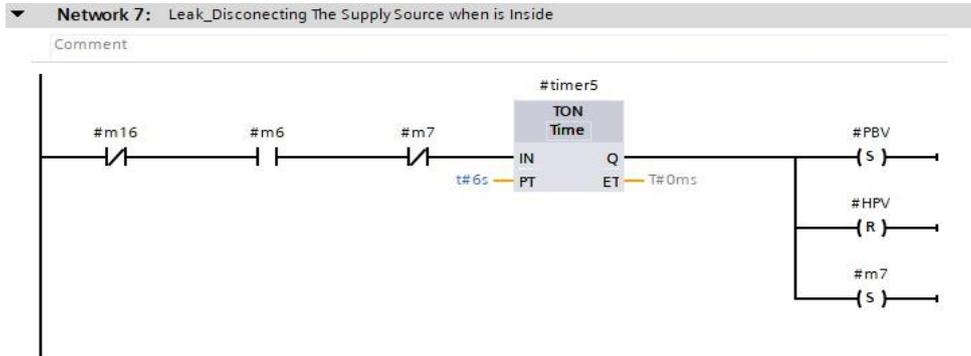
N-5- While the main supply is disconnected and the front path was open to passing the Possible leakage from the chamber to exhaust, then the main Test mode of the front chamber is started for 30 minutes. After this time the piston moves IN and goes to the next step.



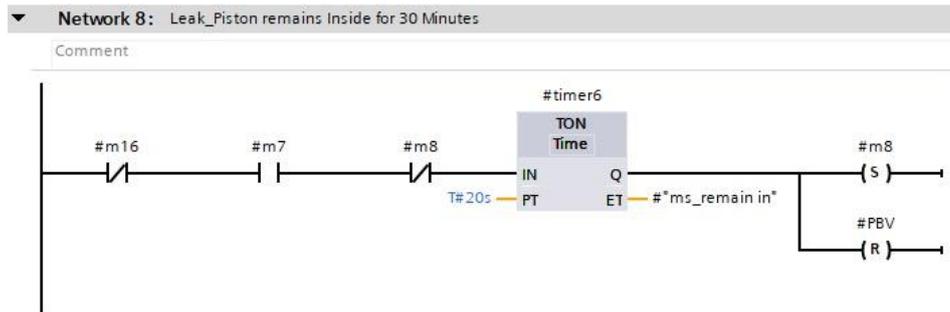
N-6- In this line piston has been moved IN and stay under pressure for sometimes, then when timer finished it disconnects the main supply and is going to another step.



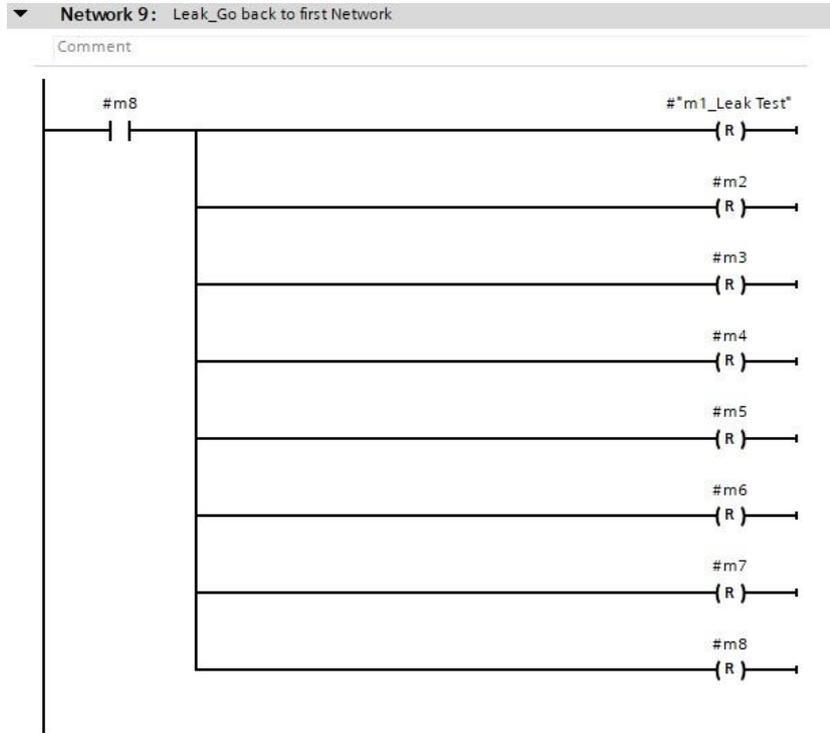
N-7- The content of this block is to keep the cylinder under pressure for a while then swap to the next network.



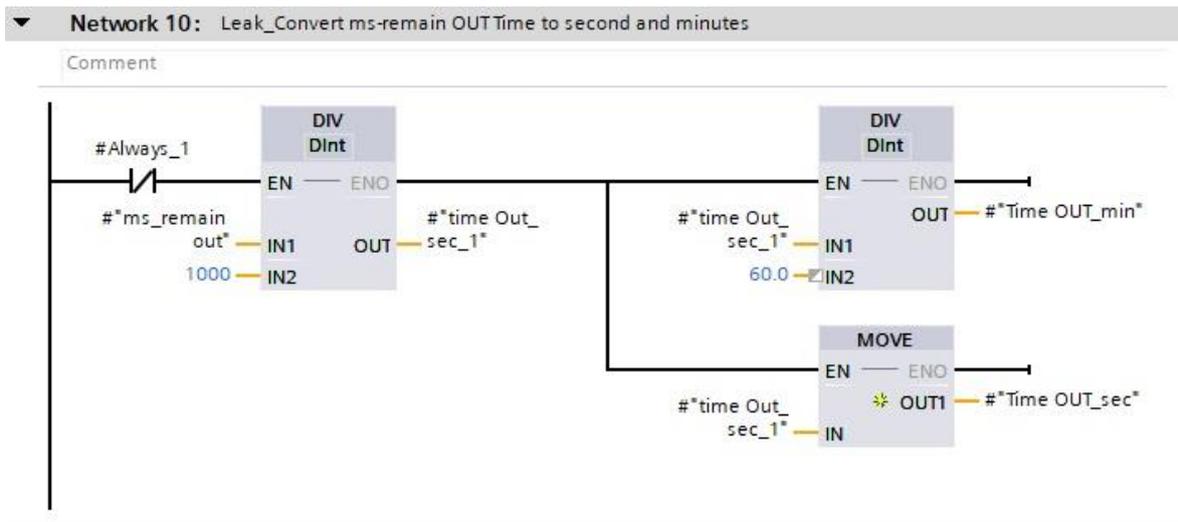
N-8- In this network, there is the main timer to start counting for the main Test of the back chamber for 30 minutes. When then 2nd test of the chamber has finished it sets a memory to send a signal for the next network.



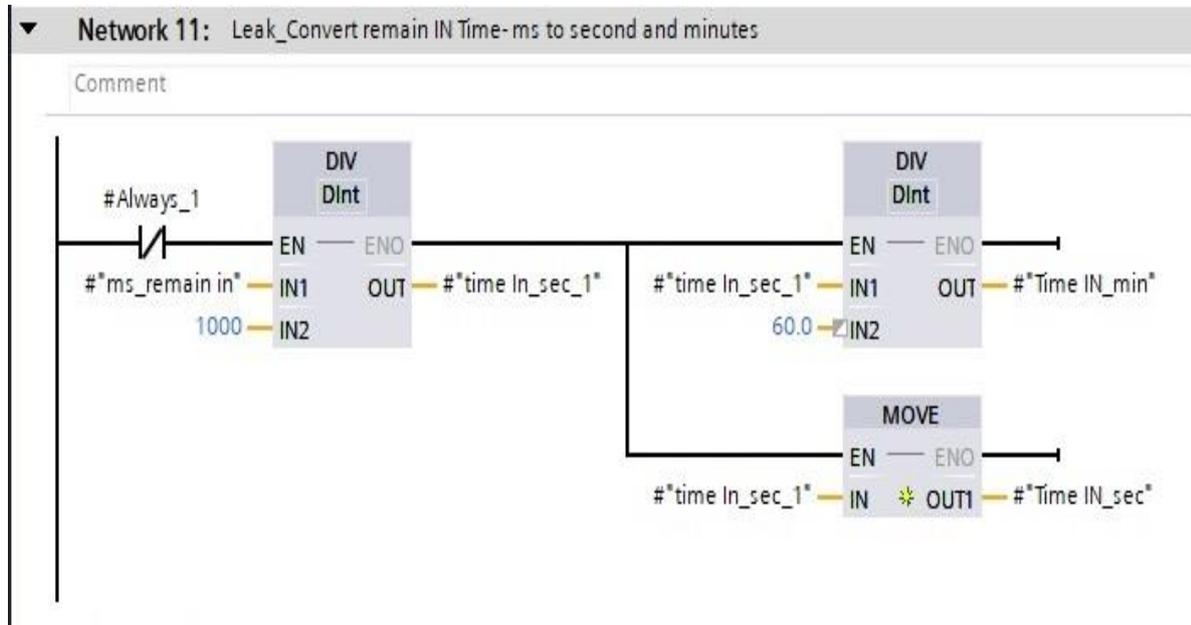
N-9- In this network by receiving a signal from the last step, the whole network of this mode will be reset, and it goes back to the first network.



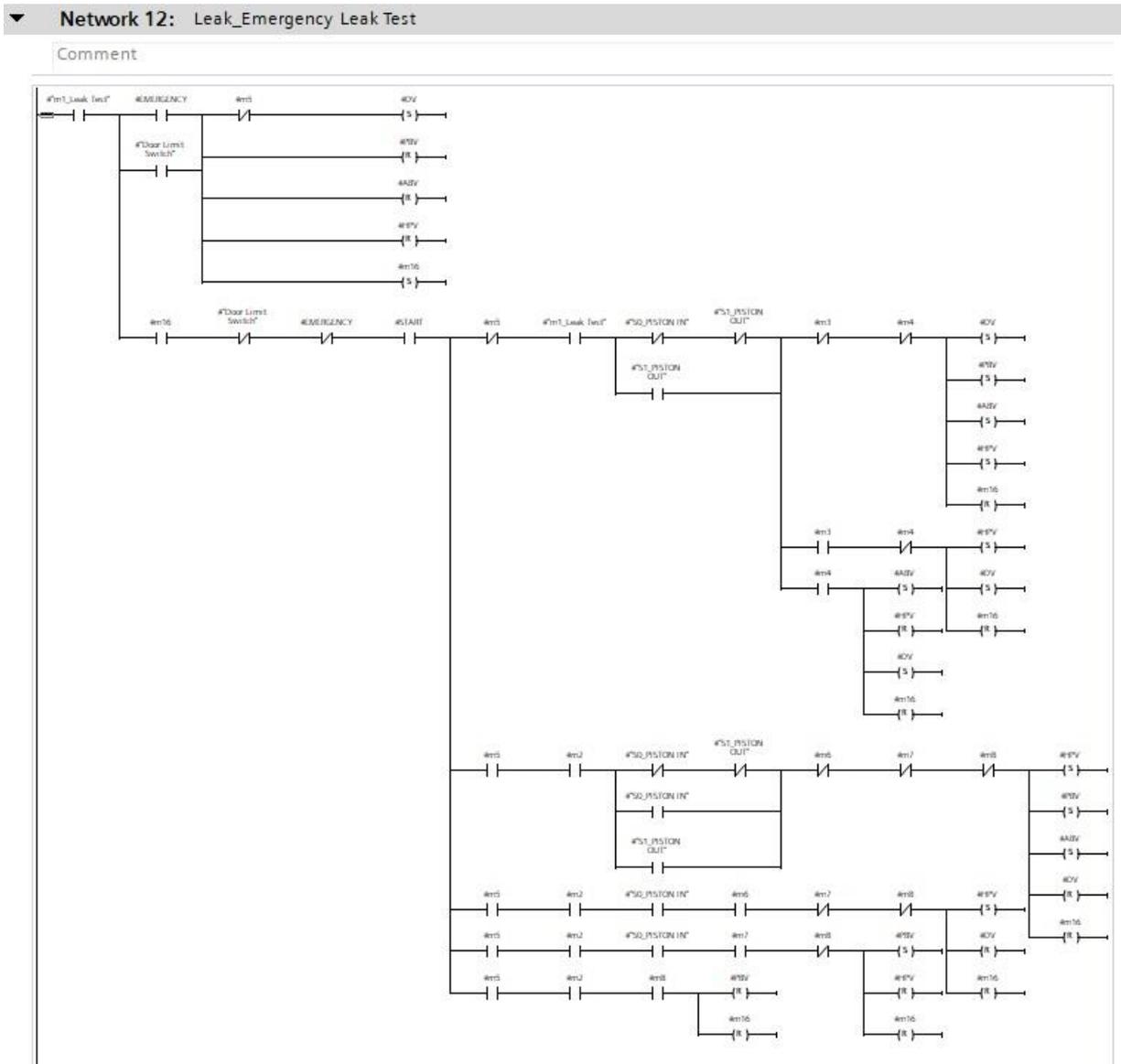
N-10- The main goal of this block is to convert the output time of timers from millisecond into seconds and minutes to use and display on the HMI panel, only for the remaining time IN.



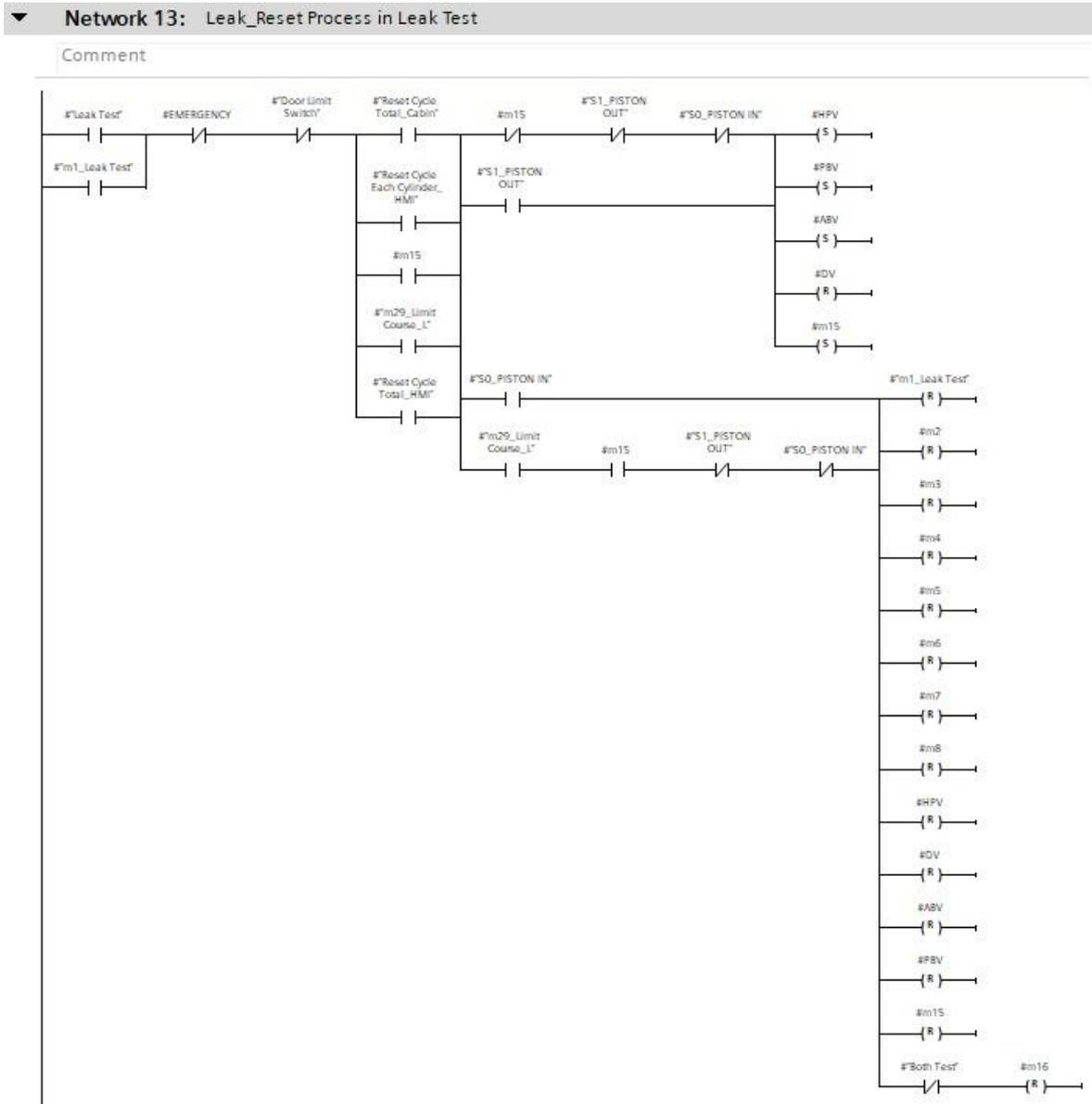
N-11- The main goal of this block is to convert the output time of timers from millisecond into seconds and minutes to use and display on the HMI panel, only for the remaining time OUT.



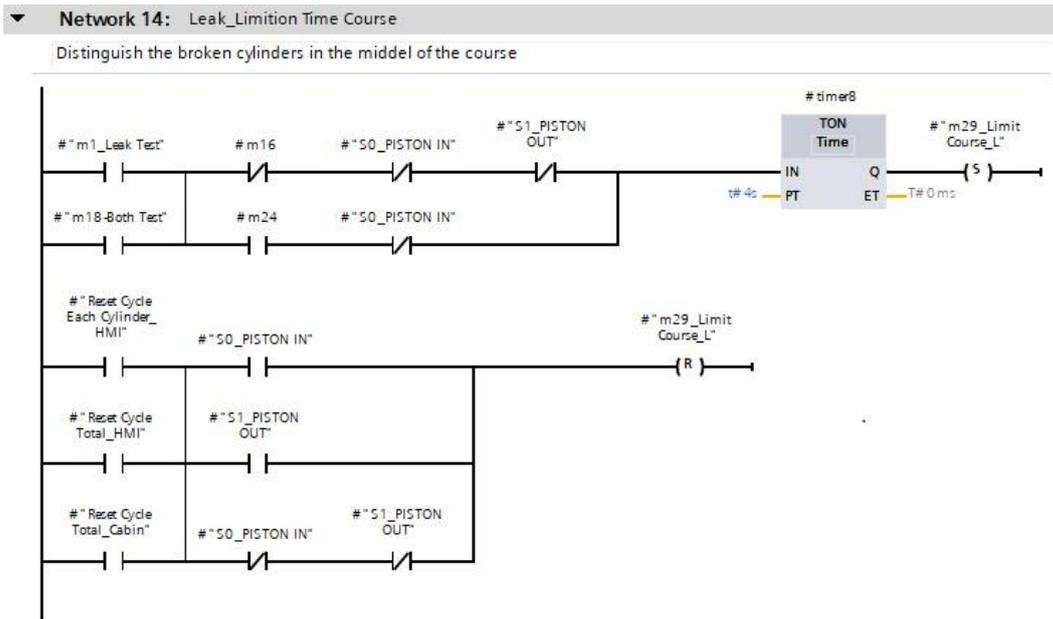
N-12- This network is playing a key role in the safety and the emergency of the Test Bench in urgent moments and is related only to the emergency sections. This network is consisting of two main parts, the first part is acting when the piston is moving out and the 2nd part is acting while the piston is moving backward. As has been explained in the previous networks, in each part there are different steps that must be considered to stop the piston exactly at the same moment of the movement to prevent the possible damages of humanity.



N-13- The functionality of this network is also very important when it is needed to reset the whole cycling in the test bench or for each cylinder cycle. This is also consisting of two main parts, and each one would be happed in two different situations and conditions. The first part is acting when the piston is moving out and the 2nd part is acting while the piston is moving backward.



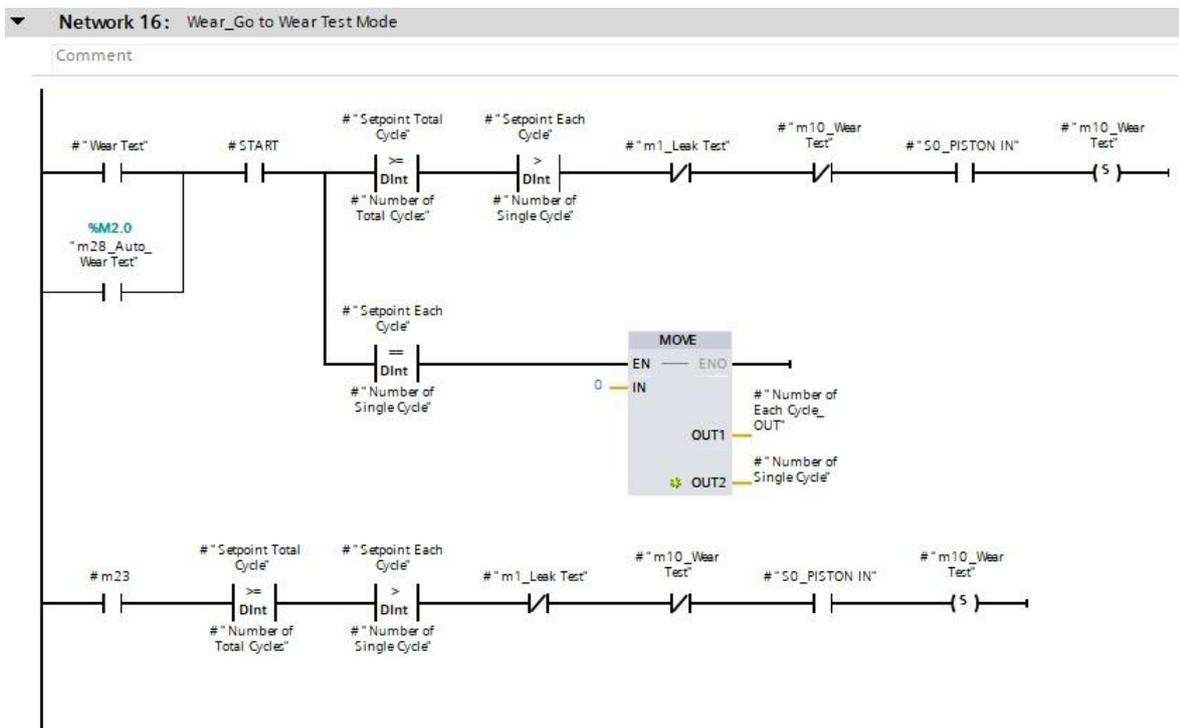
N-14- Course limitation troubles in LEAK modes, there is some probability to stop or break down the cylinders and this might happen due to sensor problems or cylinder breaking down. And it can lead to stopping all cylinders while it is working among other cylinders. So, by designing this network only the broken cylinder will be reset and go back to the initial situation and will be out of the test, in order to not disturb other processes of the test for other cylinders.



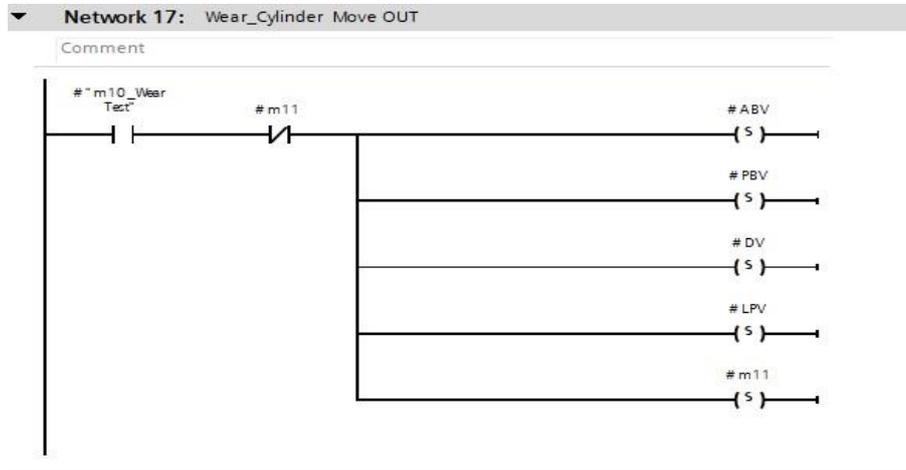
1-7-3-2- Network 16 to 27 is about “WEAR Test Mode”

N-16- In the “WEAR Test Mode”, this network is playing a central role to set the memory for commencing the WEAR test networks. There are two main separated branches, the first ladder branch is starting with two parallel contacts, which the first contact condition is coming from the physical selector mode switch on the switchgear panel that is located before start contact condition and the second one is coming from select mode switch on the HMI screen panel to set the mode.

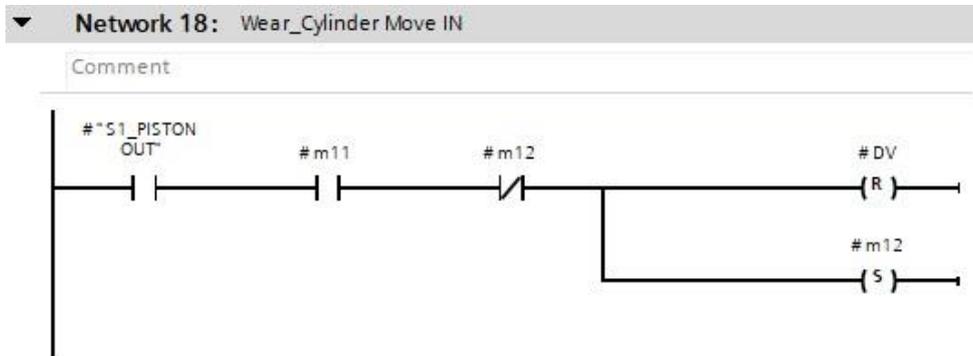
The second line of the program here would be run while both modes have been required running successively and continuously and give the possibility to run the test bench automatically.



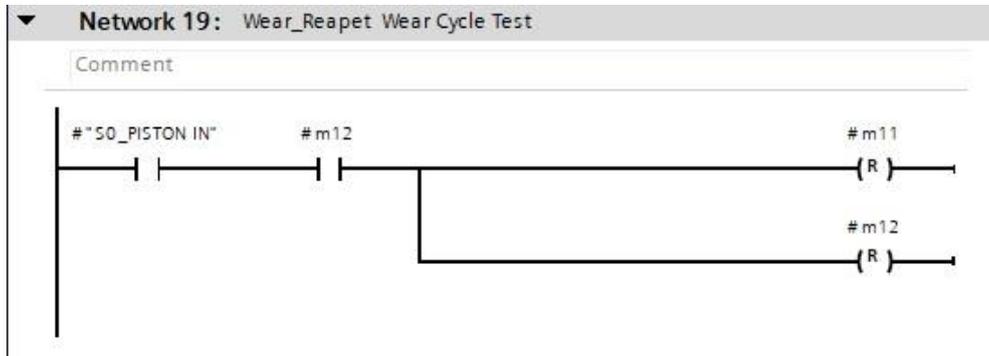
N-17- In this network, all valves which are related to moving OUT the piston, must be ON, but some conditions on their path are necessary which one of them is related to the previous network and the one other is related to memory in this network that prevents receiving a signal again.



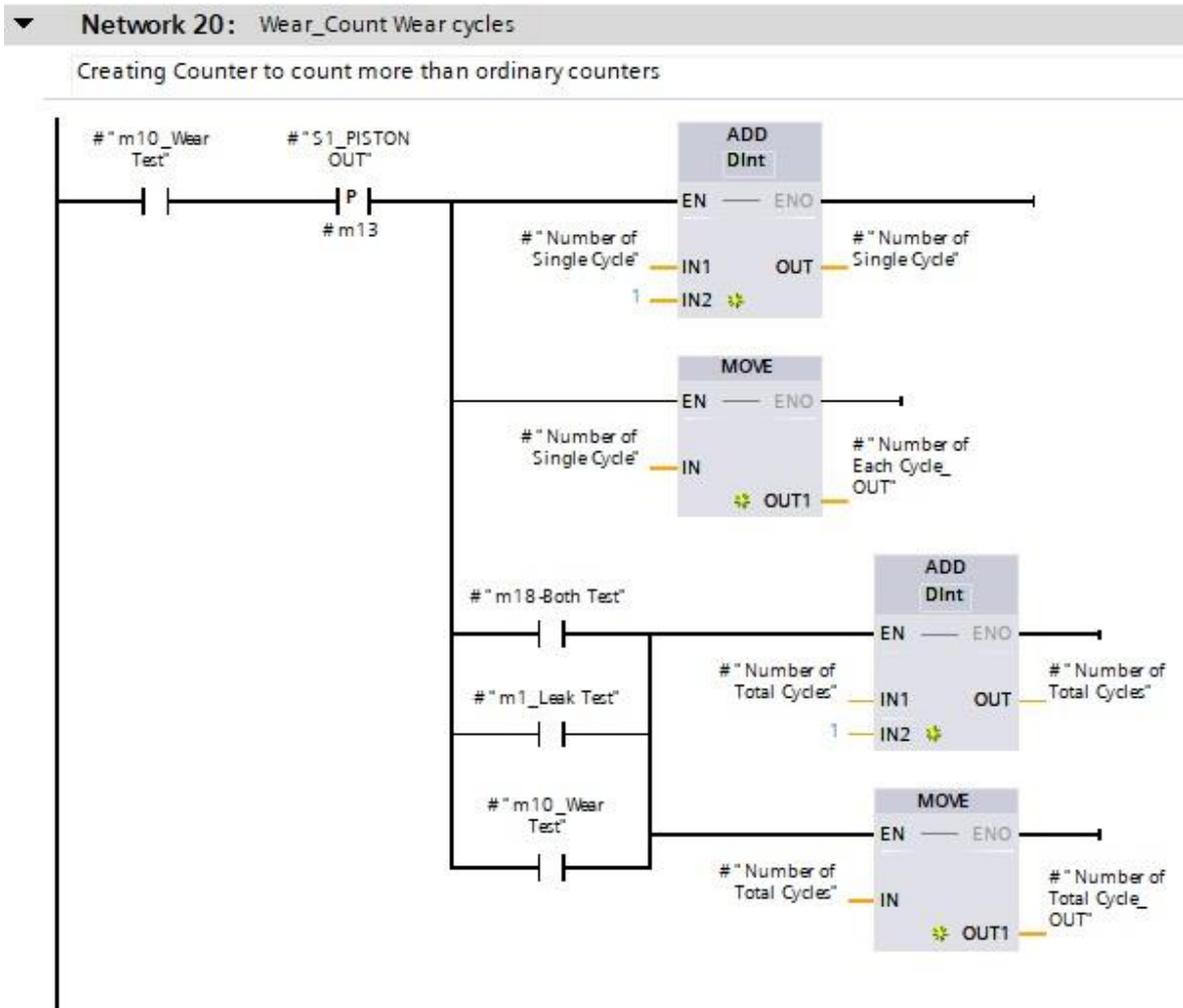
N-18- For moving the cylinder backward, this line of the program is designed, like all other networks an extra memory is set to prevent receiving a signal again and it allows the next network to be run.



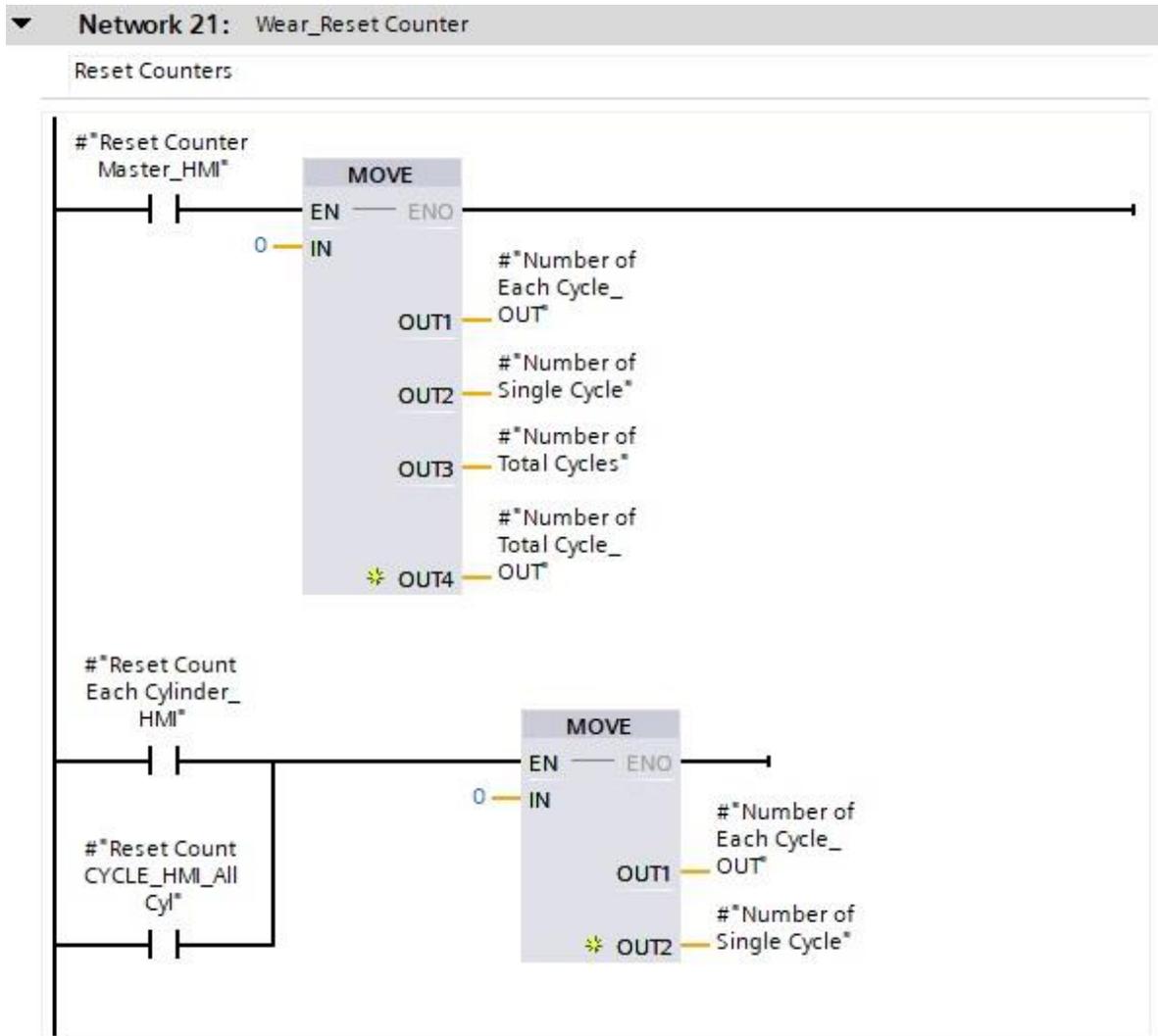
N-19- for repeating the cycles continuously and successively, this network programmed by resetting two memories which are related to moving cylinder OUT and IN, Plus receiving a signal from one of the sensors.



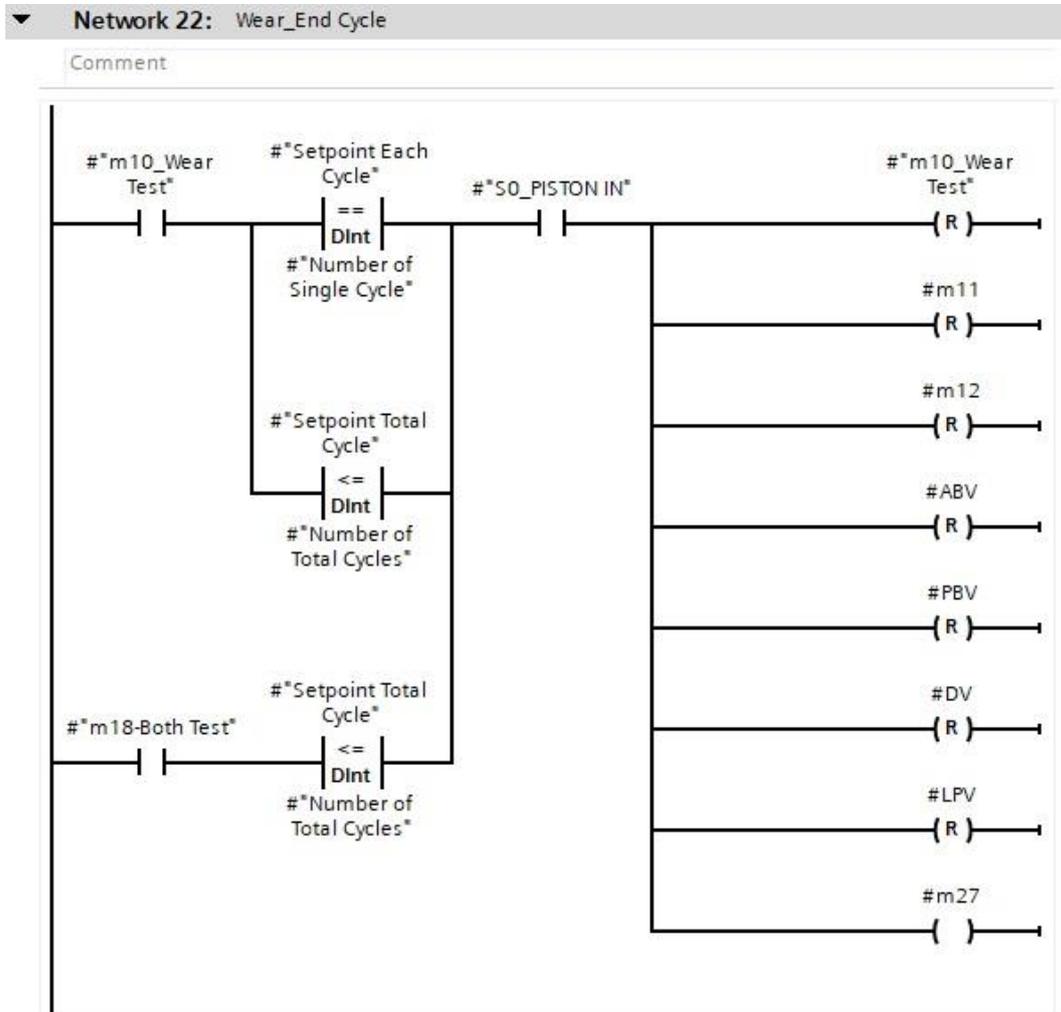
N-20- This network is about the measurement of the cylinder cycling and counting cycle. This program is counting all cycles for “each wear cycle” and “total wear cycle”. It is noted that as in this project cycles need to be counted more than a normal counter, so I have designed this program by some Adders and Move blocks for both “each cycle” and for “total cycle”.



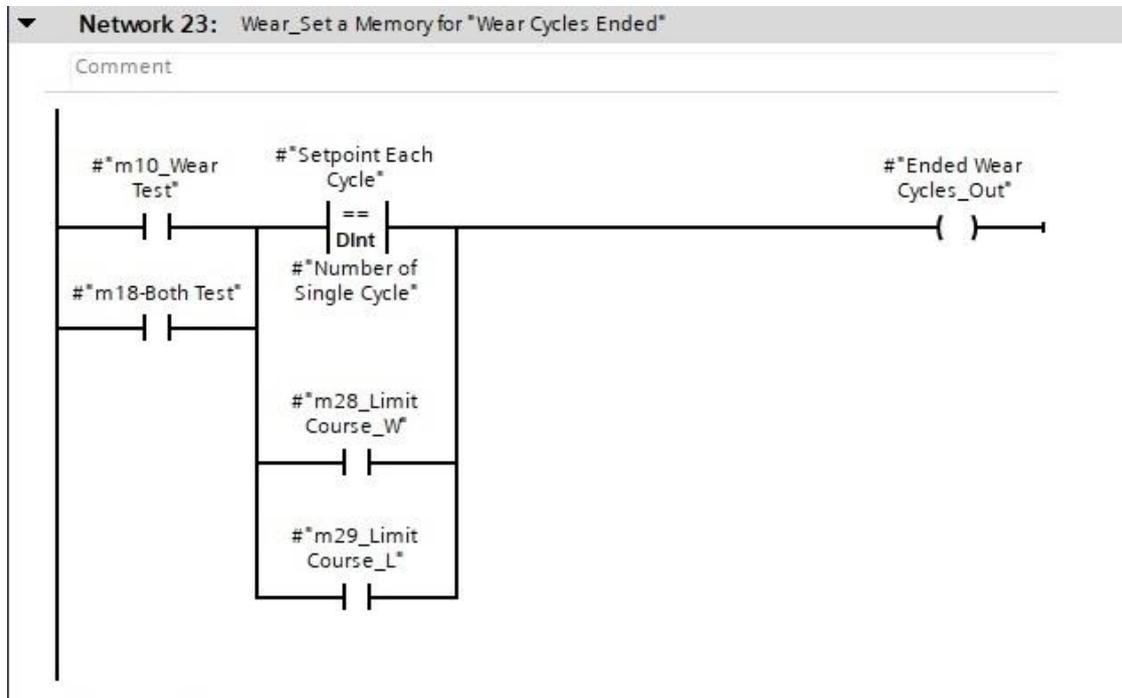
N-21- For resetting the total number of all counters together and each cycle of each cylinder, this network is used. This is working as moving a zero to all counters or counter of each cylinder.



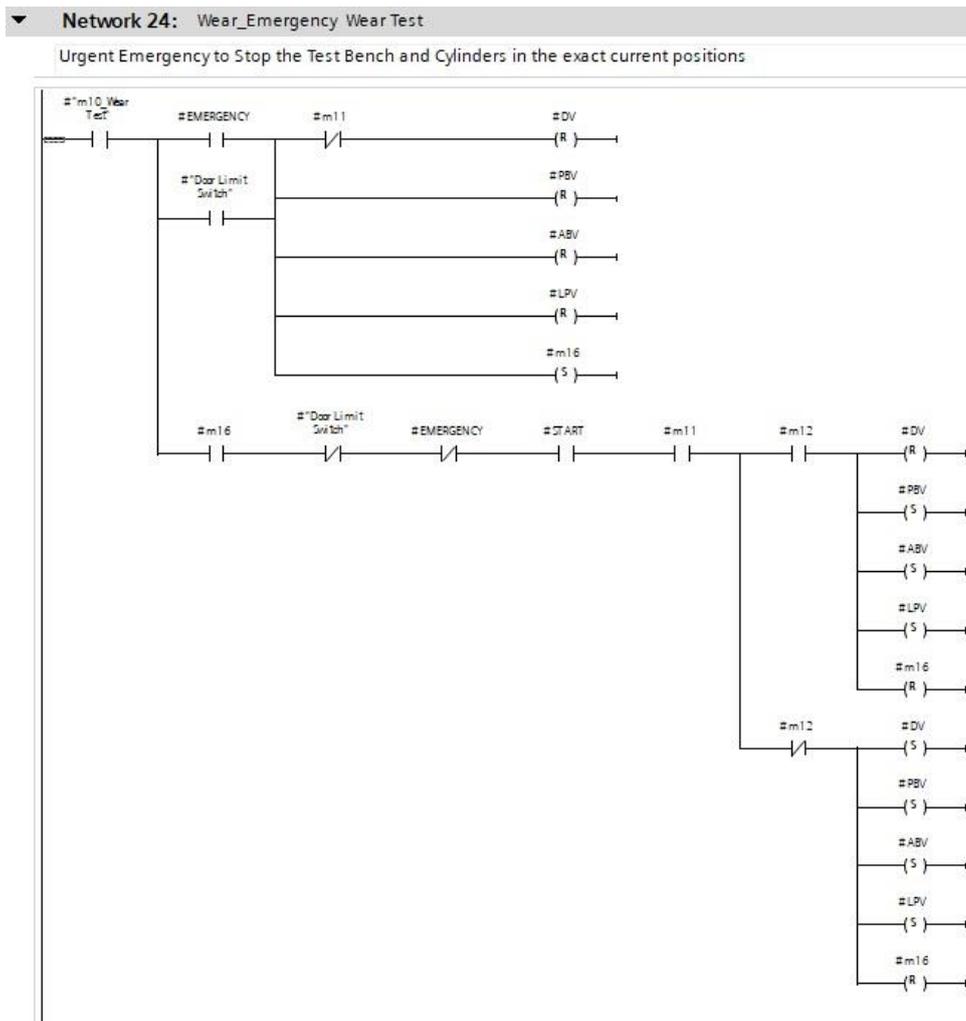
N-22- End cycle network, while the numerical conditions in both modes WAER and Both are available and the piston is in the initial situation, all valves will be reset.



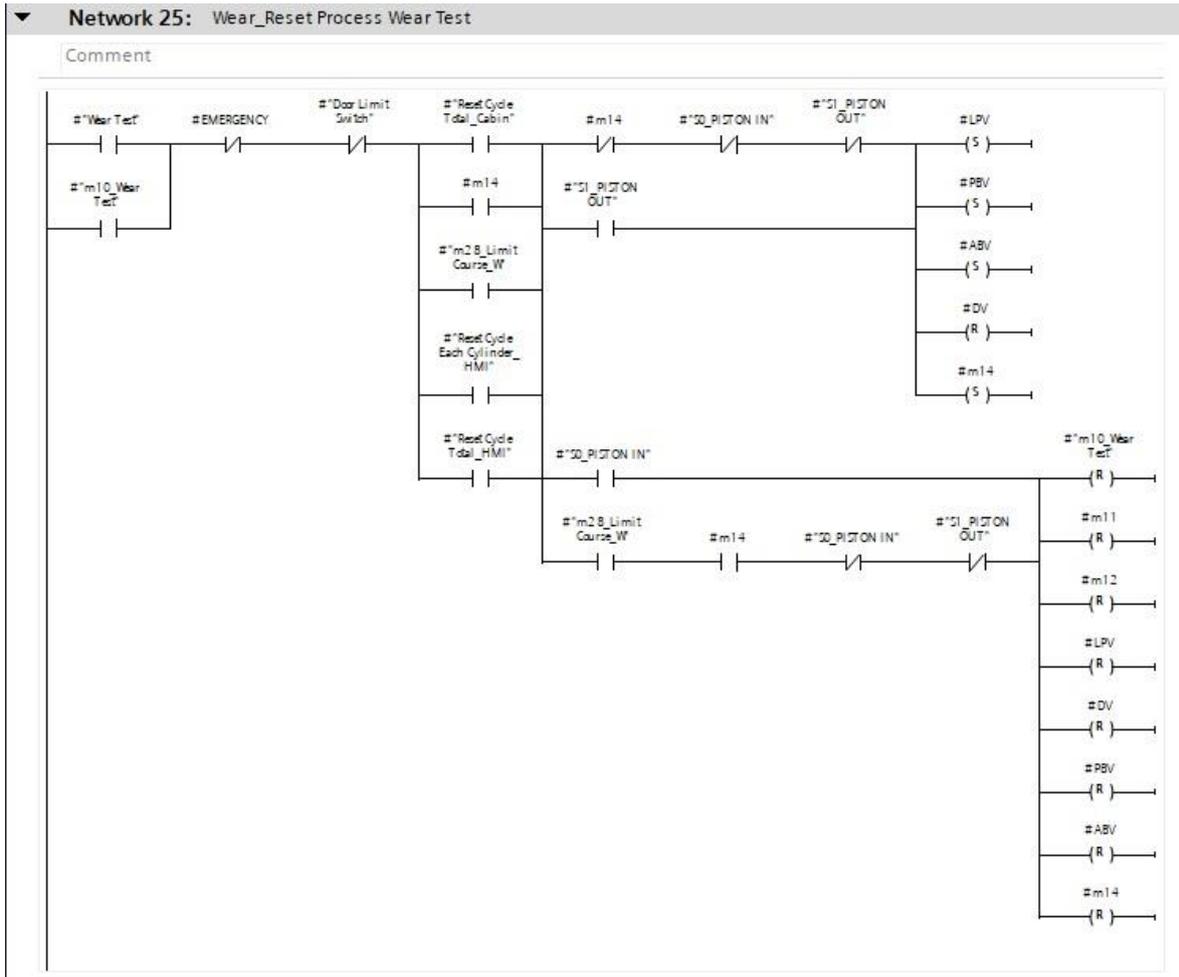
N-23- For sending a signal out of the block to use in Main OB1, this network is planned. And this signal will be used for giving the priority to all cylinders to move from one test to another test in “Both Modes Test”.



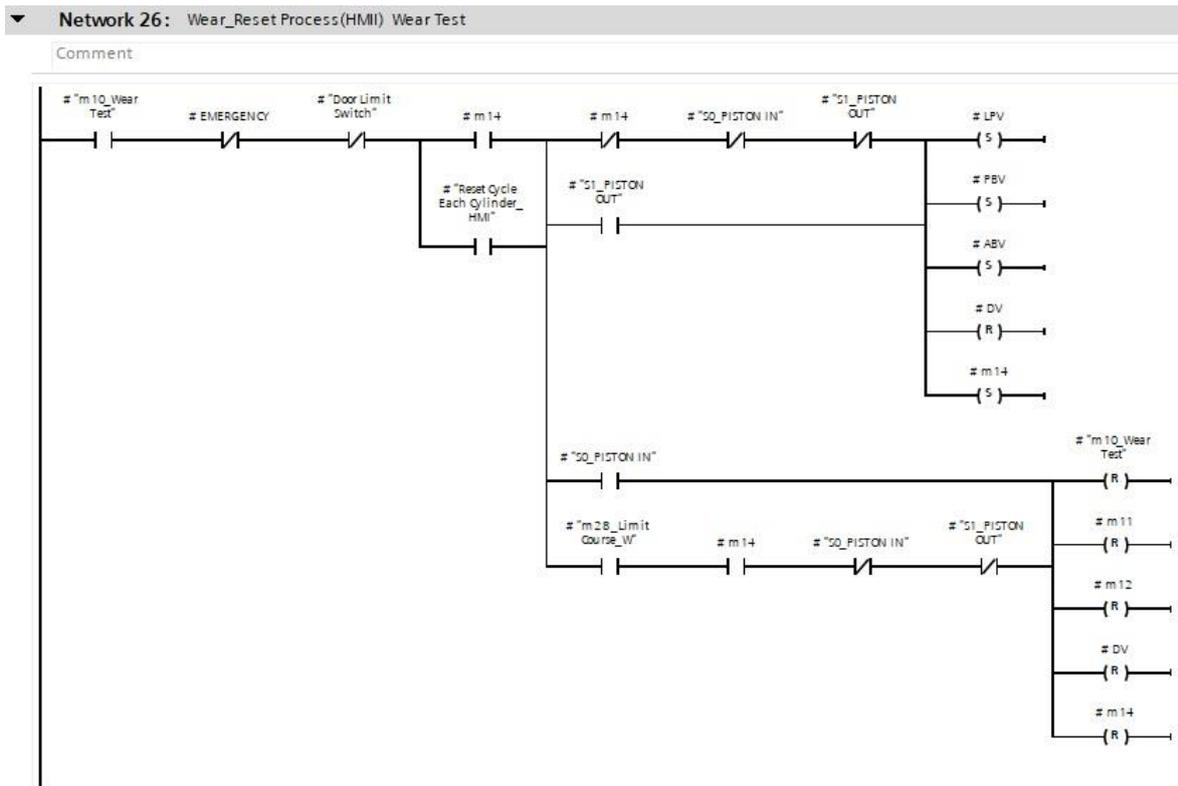
N-24- This network is playing a central role for the safety and the emergency of the Test Bench in urgent moments and is related only to the emergency sections. This network is consisting of two main parts, the first part is acting when the piston is moving out and the 2nd part is acting while the piston is moving backward.



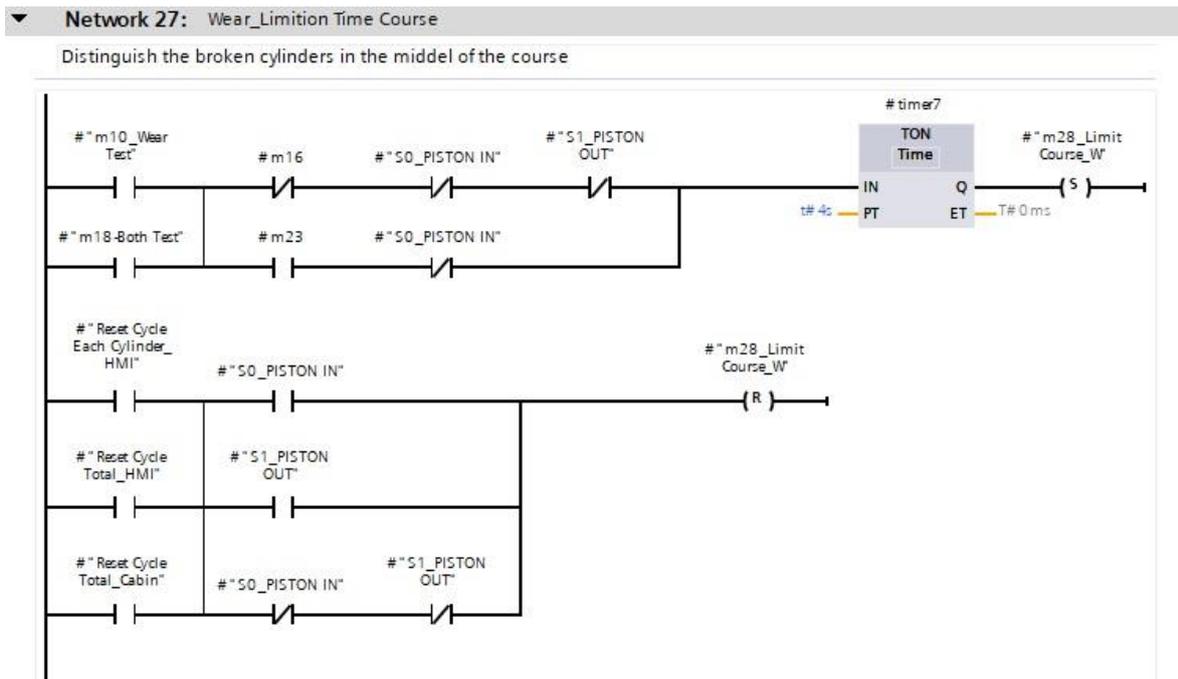
N-25- The functionality of this network is also very important when it is needed to reset the whole cycling in the test bench or for each cylinder cycle. This is also consisting of two main parts, and each one would be happed in two different situations and conditions. The first part is acting when the piston is moving out and the 2nd part is acting while the piston is moving backward.



N-26- The functionality of this network is about the Resetting the cycles from HMI display when it is needed to reset the whole cycling in the test bench. This is also consisting of two main parts, and each one would be happened in two different situations and conditions. The first part is acting when the piston is moving out and the 2nd part is acting while the piston is moving backward.



N-27- Course limitation troubles in WEAR mode, there is some probability to stop or break down the cylinders and this might happen due to sensor problems or cylinders breaking down. And it can lead to stopping all cylinders while it is working among other cylinders. So, by designing this network only the broken cylinder will be reset and go back to the initial situation and will be out of the test, in order to not disturb other processes of the test for other cylinders.

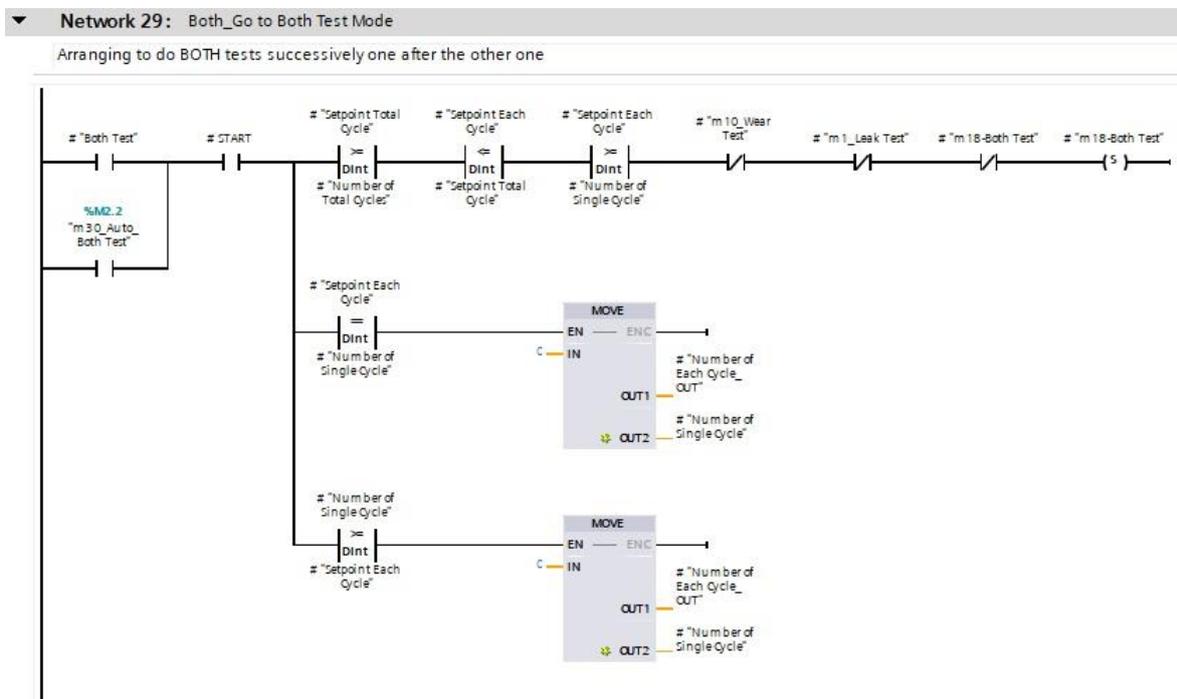


1-7-3-3- Network 29 to 35 is about “Both Test Mode”

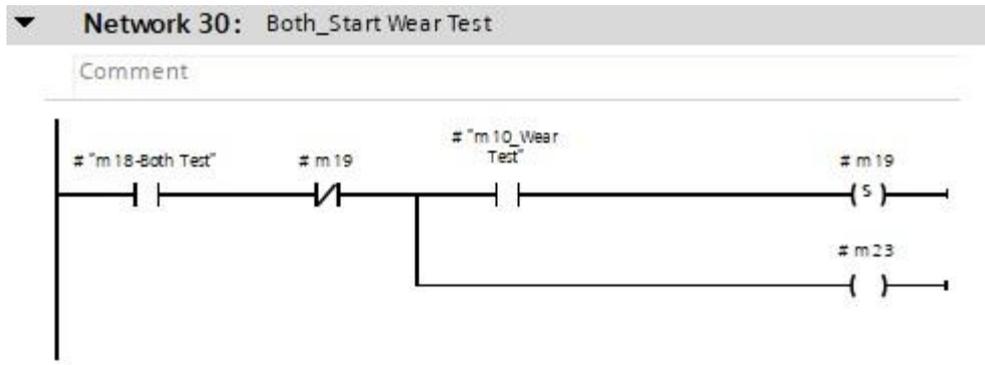
N-29- In the “Both Test Mode”, network 29 has a significant managerial role to set the memory for beginning the Both Modes Test. There are two main separated test modes, the one is WEAR and the second one is a LEAK test. In this mode, they must be run successively and continue until reaching the number of total cycles setpoint.

There are two parallel contacts at the beginning of the network. The first contact condition is coming from the physical selector mode switch when neither of the two modes is available, from the switchgear panel and the second one is coming from select mode switch on the HMI screen panel to set the mode.

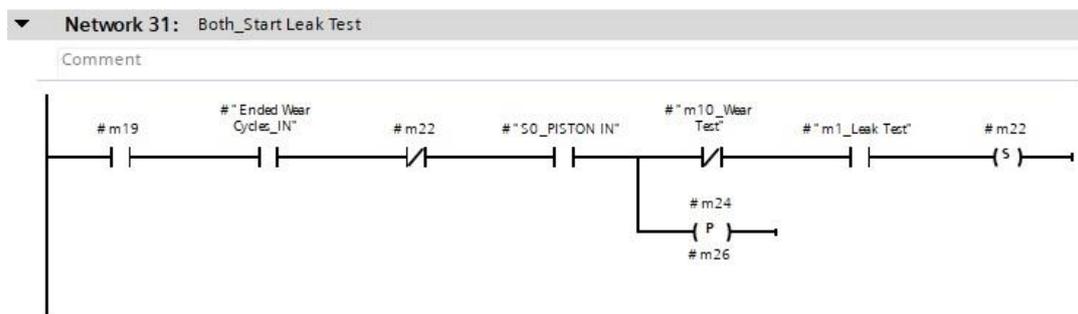
There are two Move blocks, which sand zero to Counters while the “number of each cycle out” is equal to the “setpoint each cycle” to allow the test bench to run again without pushing the reset button.



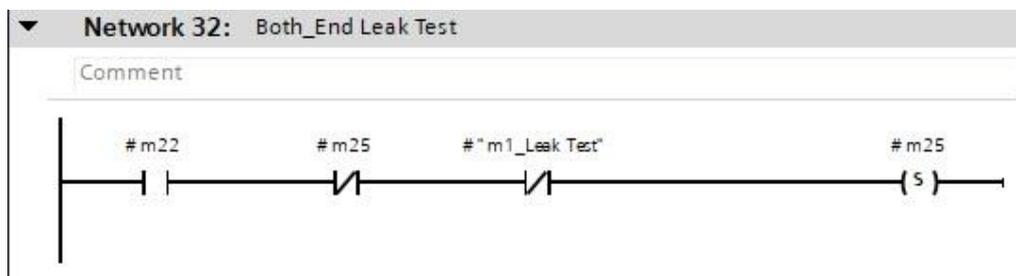
N-30- Start WEAR test automatically in this mode, now is the time to send only a signal to run the WEAR test with this network.



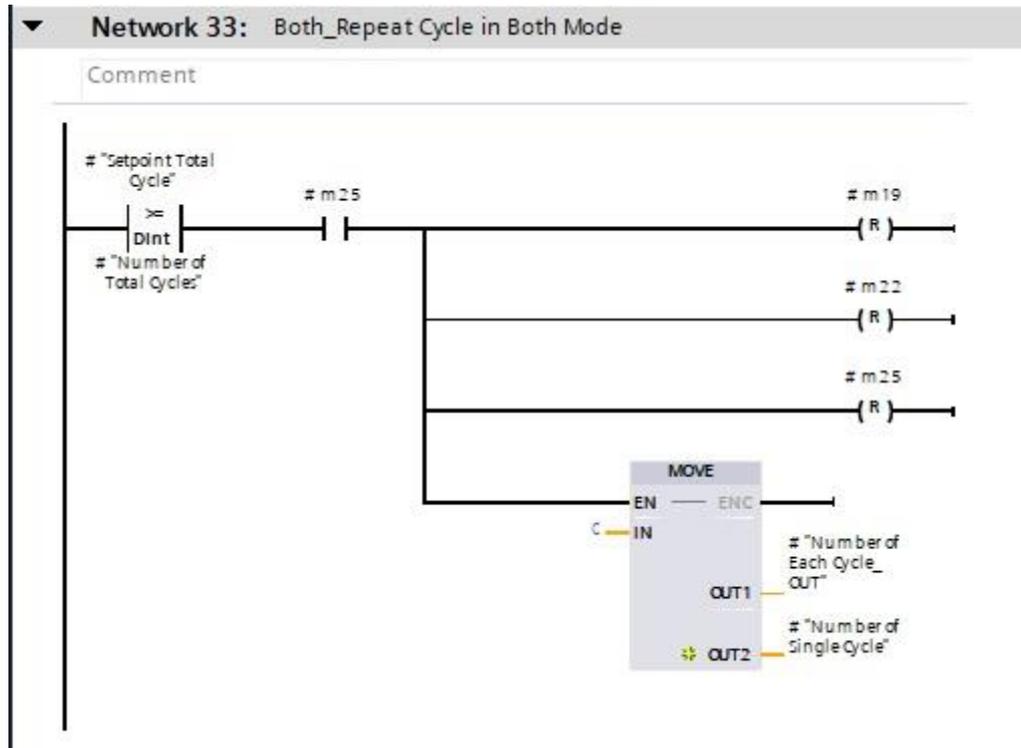
N-31- This network has a duty to run then leak test automatically, there is a very important condition in the programming line which is called “ended we're cycles_IN”. This memory would be activated while all cylinders in the WEAR mode finish then this memory allows activating the LEAK mode for all cylinders at the same time and will send a pulse to run the Leak test mode.



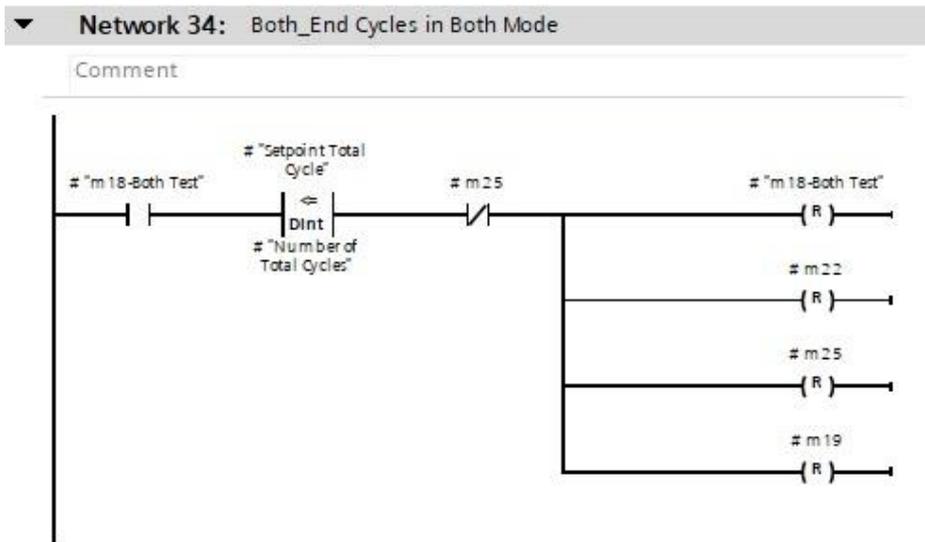
N-32- In this network, the memory will be set, while the LEAK is Finished. Also, it sends a command to the next step.



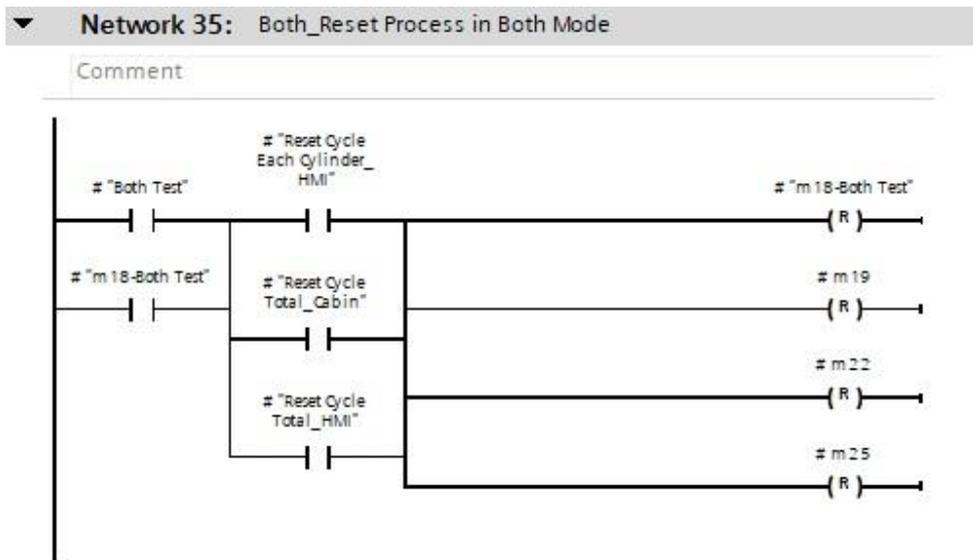
N-33- In this network, a numerical condition has been used to repeat the cycle until the condition is valid. while the “setpoint total cycle” is bigger or equal with “number of total cycles” this condition allows us to reset the memories and commence running this mode again and again. When this condition is not available compiler goes to another network to execute that one.



N-34- To finish the Both modes test, a numerical condition is used. If the “setpoint total cycle” is smaller than or equal with “number of total cycles” this network reset all related memories and all the test modes had done.

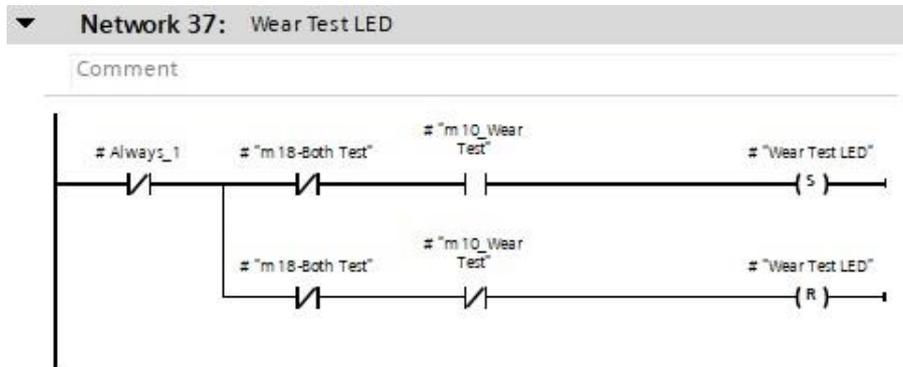


N-35- This network is related to the reset process in both modes, that can receive a signal from “reset cycle each cylinder_HMI”, “reset cycle total_cabin” or “Reset cycle total_HMI”. But it works only in this mode and reset only memories in this Mode.

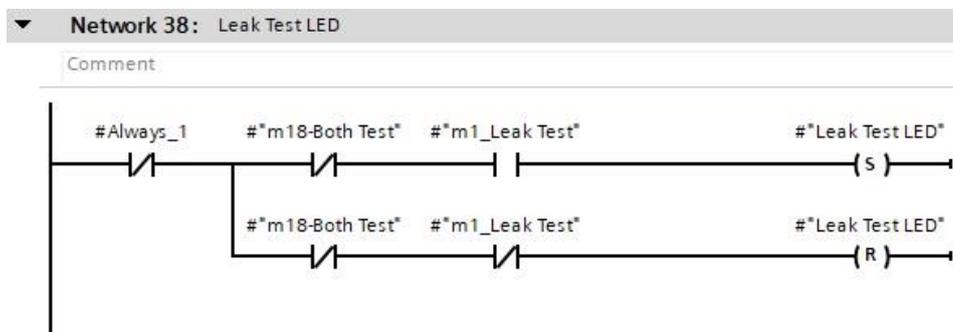


1-7-3-4- Network 37 to 41 is about “LED Output Indicators”

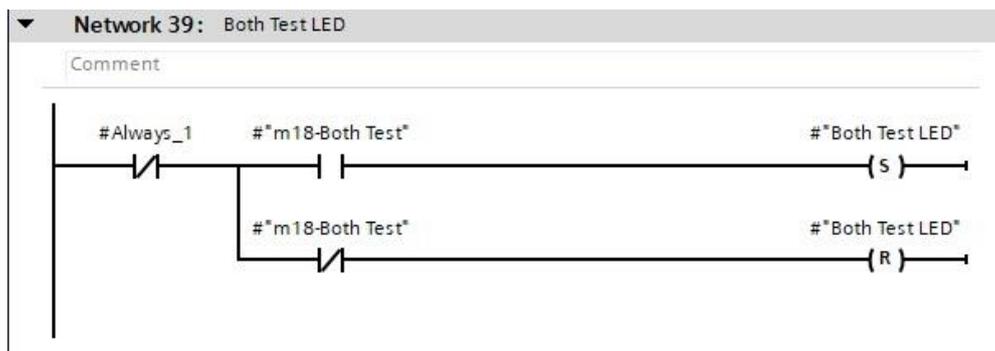
N-37- In this network Wear test LED has received the commands.



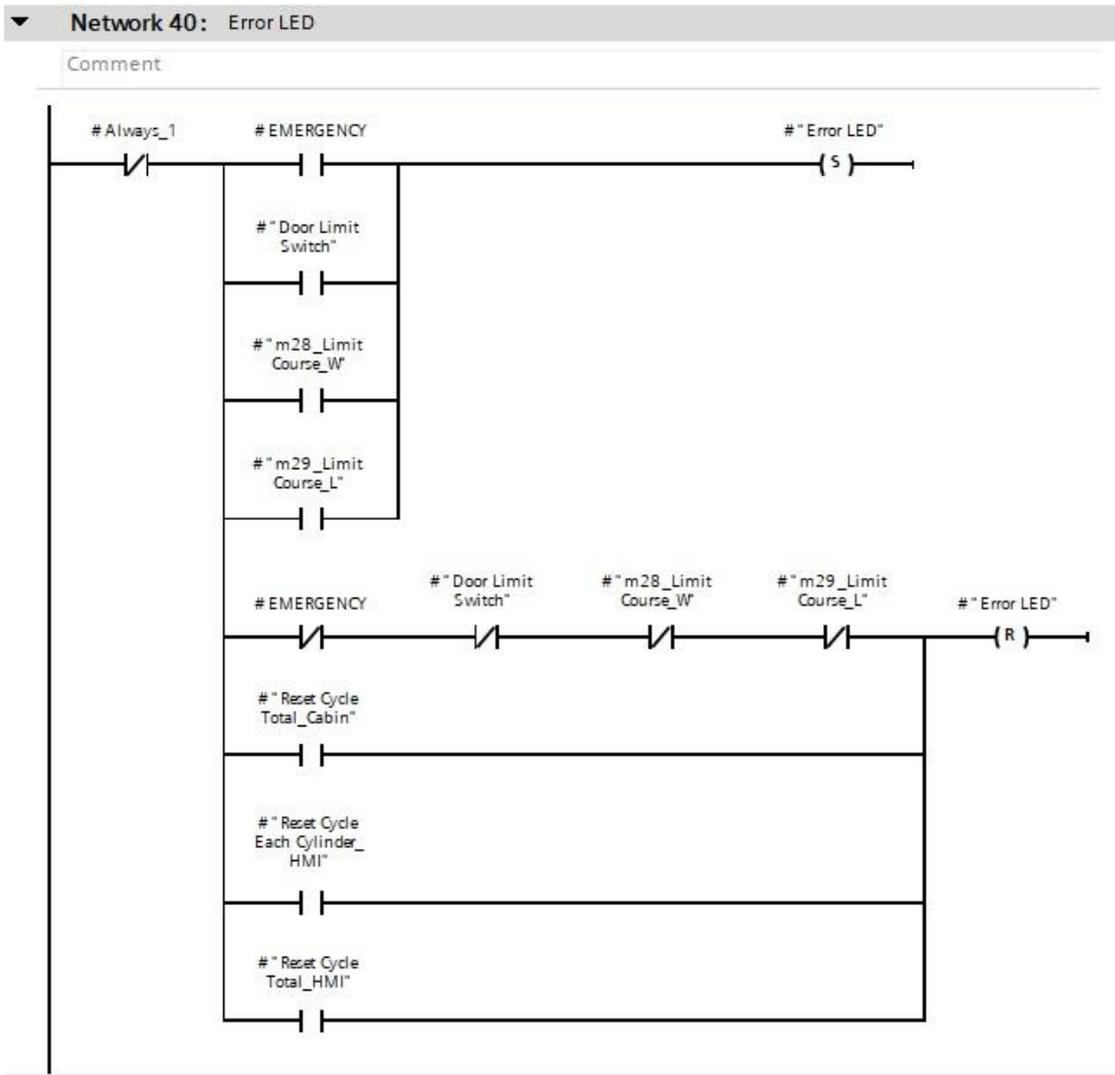
N-38- In this network LEAK test LED has received the commands.



N-39- In this network Both test LED has received the commands.

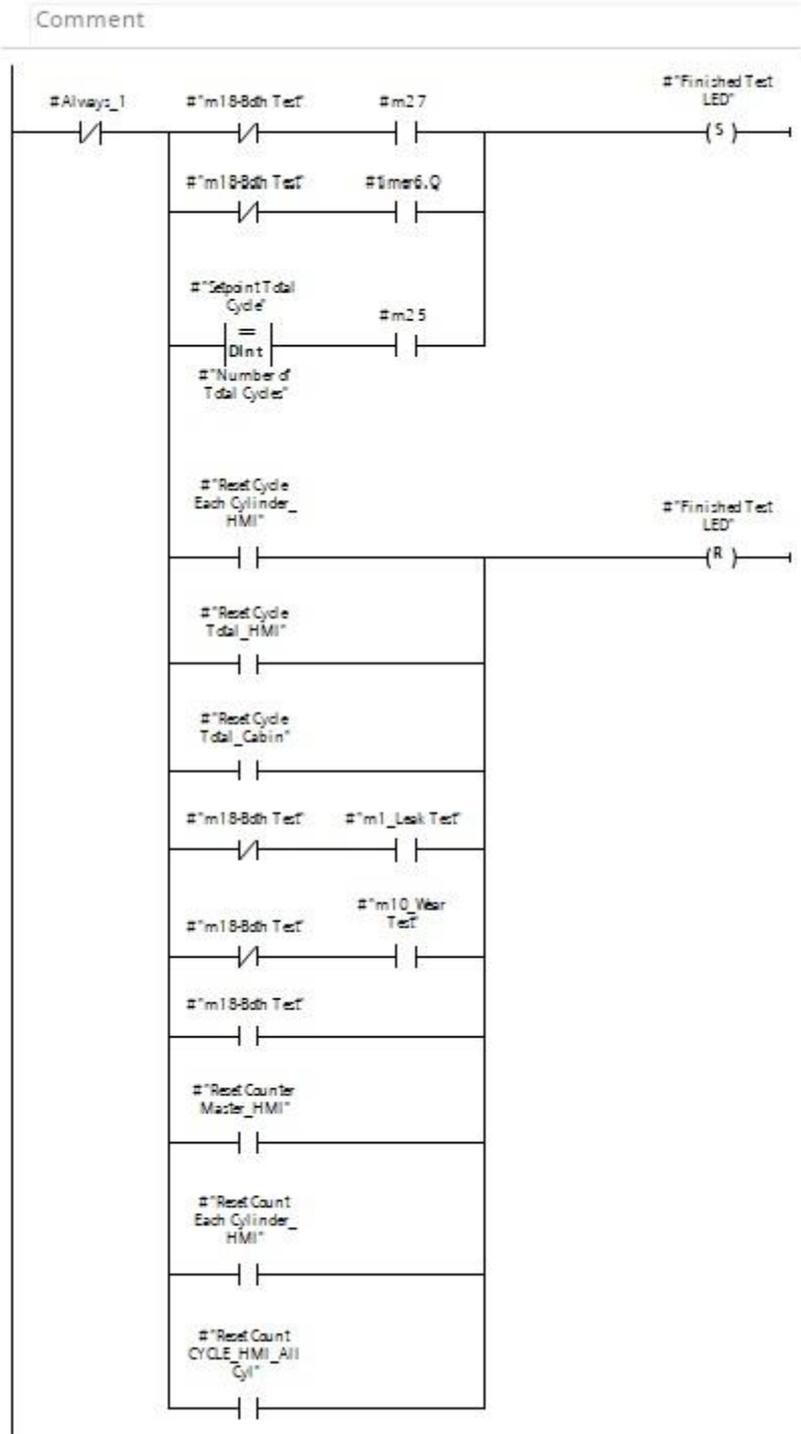


N-40- In this network Error test LED has received the commands.



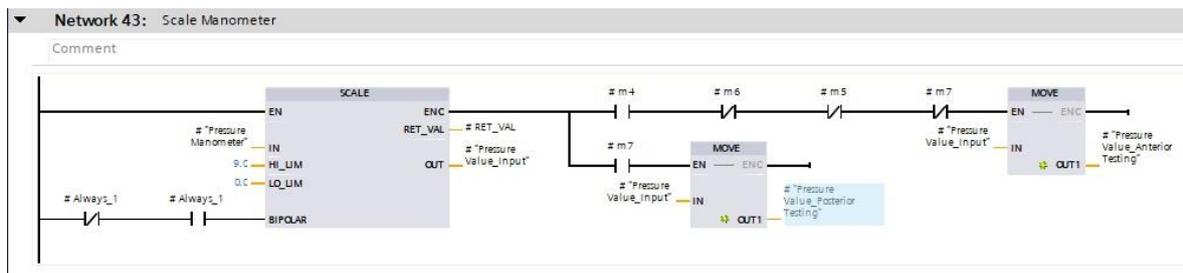
N-41- In this network Finished test LED has received the commands

Network 41: Finished Test LED



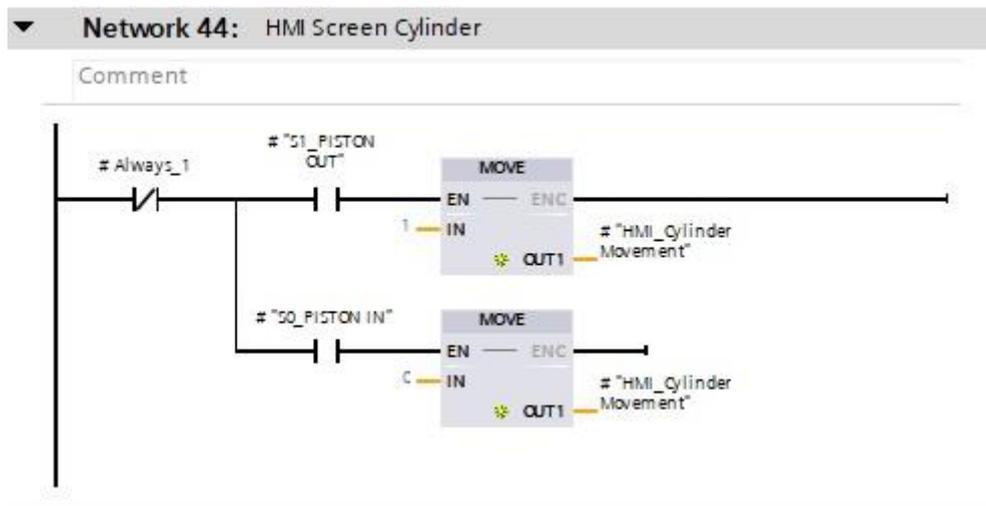
1-7-3-5- Network 43 to 45 is about 3 different part

N-43- The most important part of the program, analogue scaling, has performed in this network. After scaling the analogue input, it is necessary to move it to two different parallel directions for allowing us to put some essential conditions on the path of enabling the Move Block. The output of these blocks is used to save them in memory as a text file and allows us to use them for showing the exact number of the analogue input signal on the HMI panel.

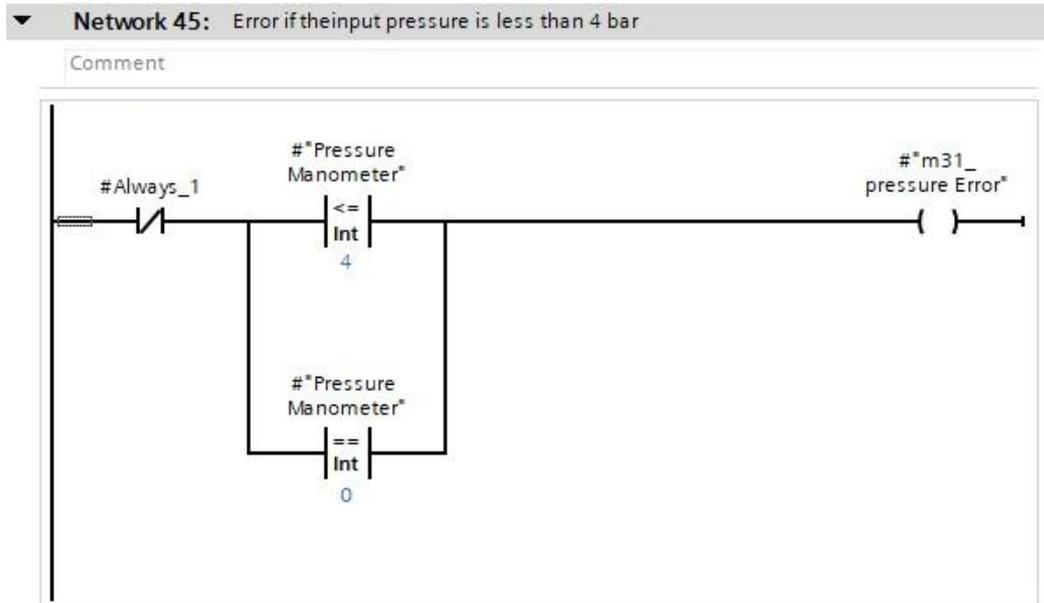


N-44- This networking is doing a beautiful action for the HMI screen. The movement of the piston on the HMI can receive the data from this network to show the action of the animation movement of cylinders.

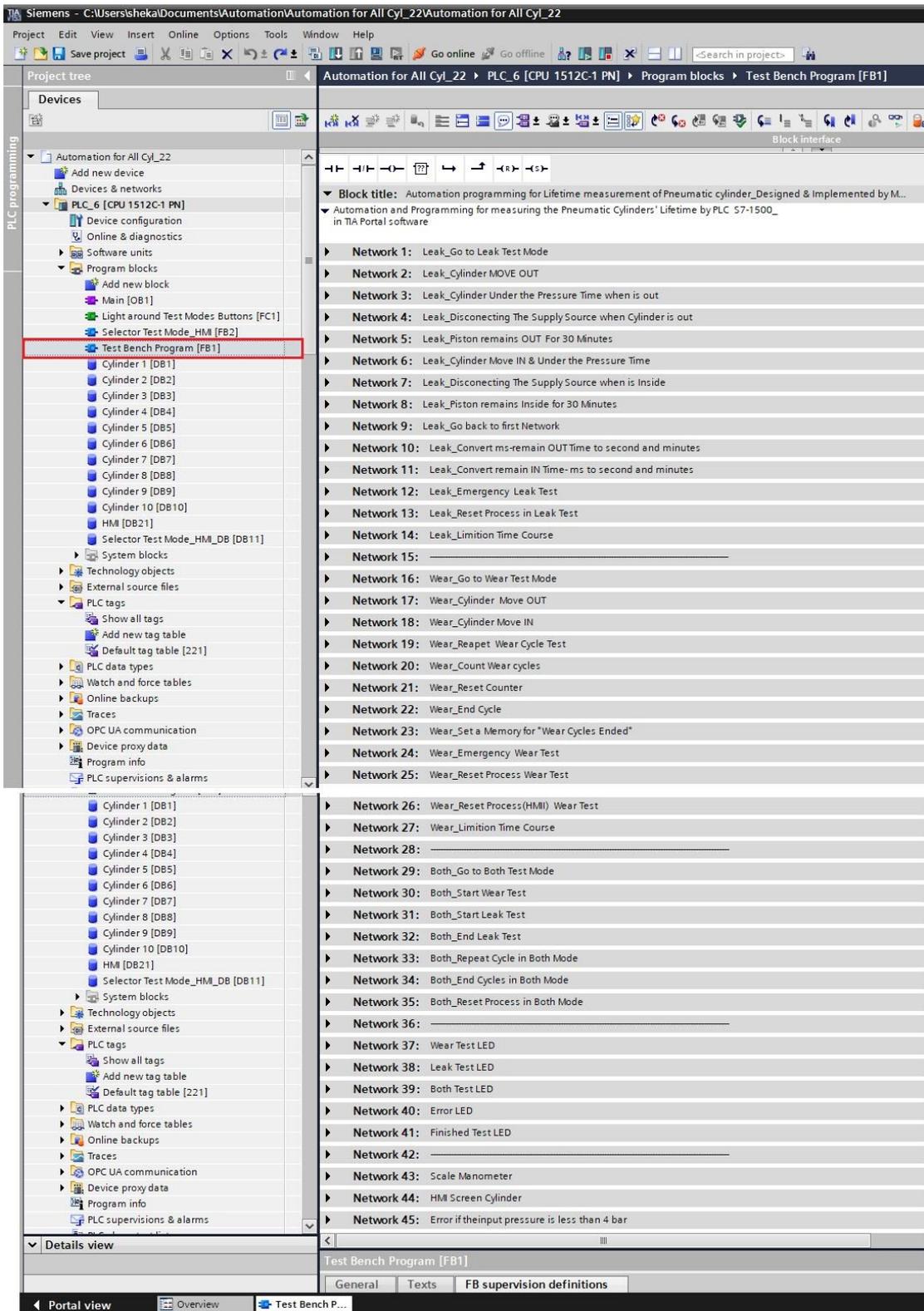
By enabling two Move Blocks with two sensors, S1 and S0, moving a 0 and a 1 to a memory, the animation movement is possible.



N-45- In the last network there are two numerical conditions for air pressure. If air pressure decrement is less than 4 bar or 0 bar the pressure Error will appear on the Error LED and it will remain ON to indicate there is a problem in the test bench air supply.



All networks inside the Function Block [FB1] shown in the figure below.



Figure

1-7-4-Selector Test Mode Function Block - For HMI [FB2]

For this test bench, a basic HMI panel is available, which can be configured by TIA portal V15.1 software. But in the library of this software, the 3-position selector switch is not available to call on the HMI screen and configure. So, I have designed a program block for three buttons to create a 3-position selector switch on HMI display, however, it is extendable for more buttons. This block has 4 input signals and 3 output signals.

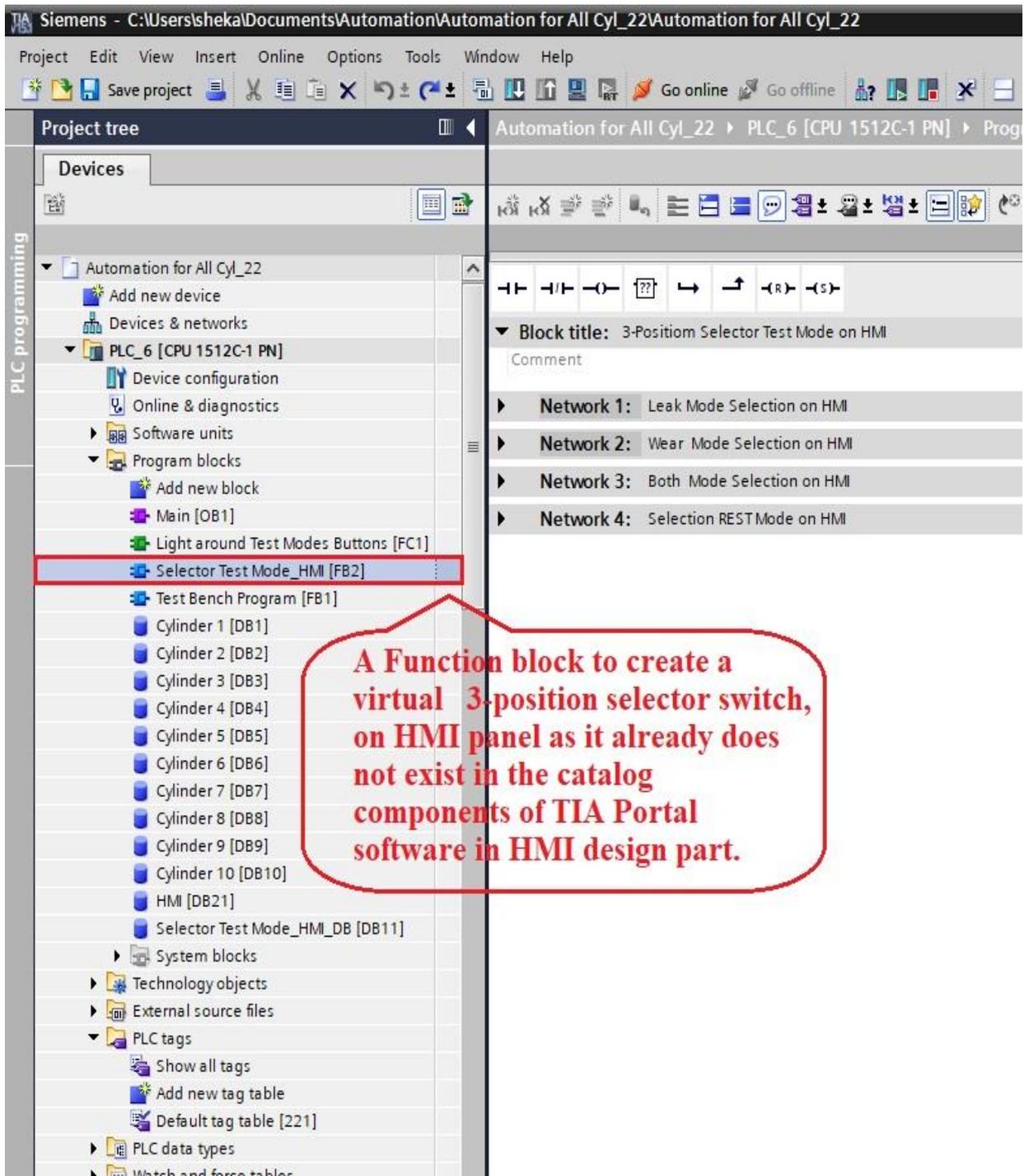
In this function block, there are four networks, that I will explain one by one.

Network 1- In this network, one of the modes will be set, and two other modes will be reset. By sending a signal, from one of the three designed buttons on HMI to the first input of the block, this mode will remain set in the output of the block.

Network 2- In the second network, one of the other modes will be set, and two other modes will be reset. By sending a signal, from one of the three designed buttons on HMI to the second input of the block, this mode will remain set in the output of the block.

Network 3- In the 3rd network, the last mode will be set, and two other modes will be reset. By sending a signal, from one of the three designed buttons on HMI to the 3rd input of the block, this mode will remain set in the output of the block.

Network 4- The last, but not least, this is about resetting the block while it is on Auto mode state on HMI panel. While this input receives a signal by pushing one of the reset buttons or changing the Auto-Manual switch on HMI, this block will be reset. It is noted that if you want to change the test mode, 1st whole block must be reset and then choose another mode to run the test.



Figure

1-7-5-The Light around Test Mode Buttons on HMI [FC1]

This block has been designed for indicating the Modes on HMI panel by activating three indicator lights (LEDs) behind the 3-position selector switch mode. All the program about Light Mode is written in a block “Function [FC1]” that does not need Data Block for storage the data.

In this Function, there are four networks, that I will explain all.

Network 1- In this network, one of the light modes will be set, and two other light modes will be reset. By receiving a signal, from one output of the 3-position selector Block, to the first input of the block, this mode will remain set in the output of the block to show it on the HMI.

Network 2- In the second network, one of the other light modes will be set, and two other light modes will be reset. By receiving a signal, from one output of the 3-position selector Block, to the second input of the block, this mode will remain set in the output of the block to show it on the HMI.

Network 3- In the 3rd network, the last light mode will be set, and two other light modes will be reset. By receiving a signal, from the last output of the 3-position selector Block, to the 3rd input of the block, this mode will remain set in the output of the block to show it on the HMI.

Network 4- Finally, this block is about resetting the light modes while it is on Auto mode state on HMI panel. While this input receives a signal by pushing one of the reset buttons or changing the Auto-Manual switch on HMI, this block will be reset. It is noted that if you want to change the test mode, 1st whole block must be reset and then choose another mode to run the test.

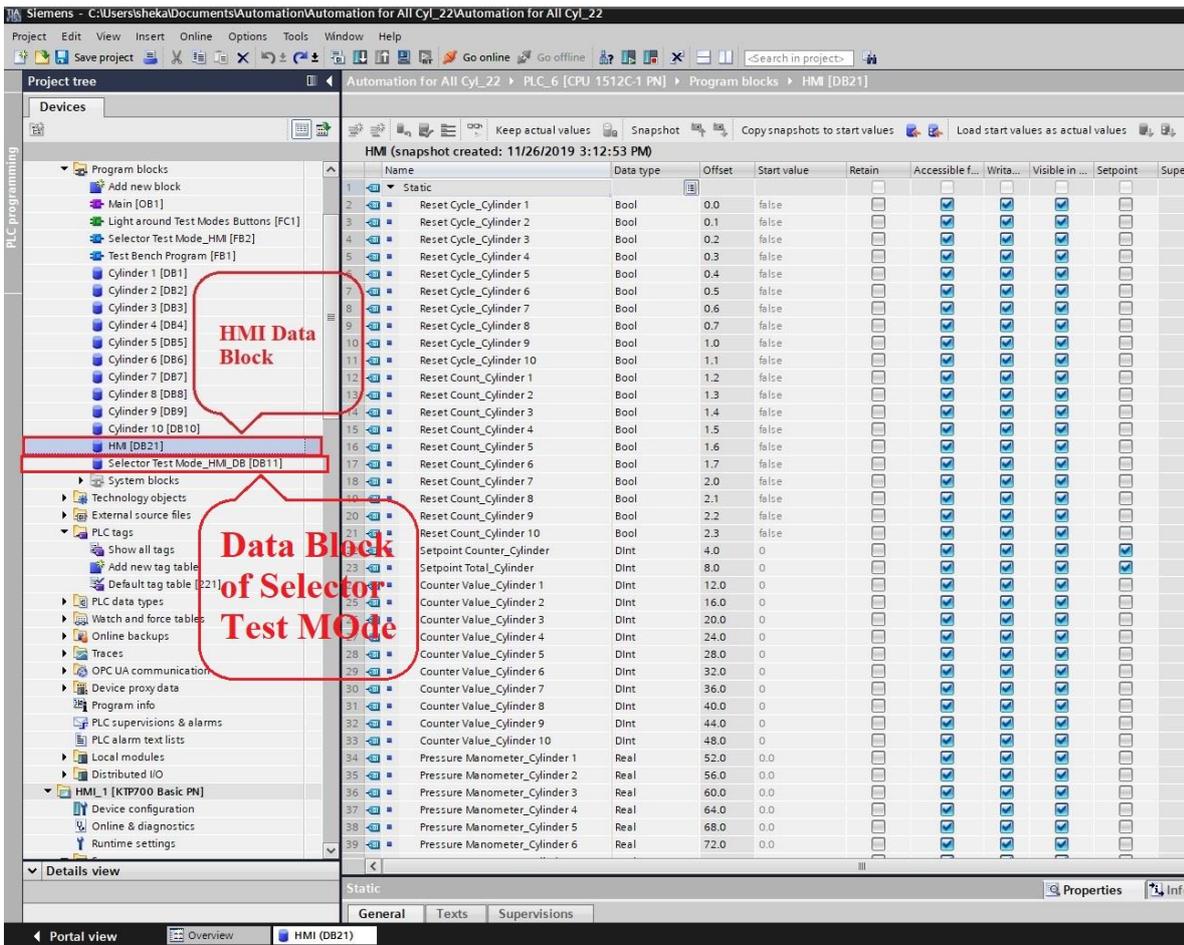
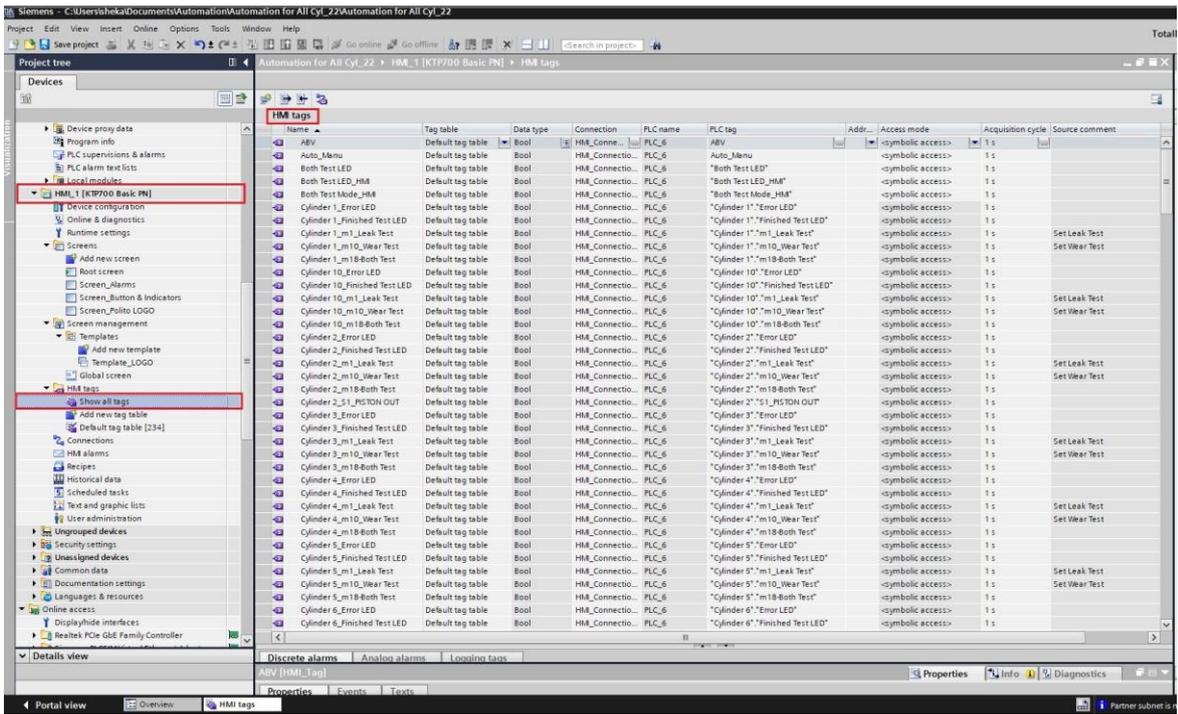
B. 1-8- HMI Conflagration and Design

Nowadays in many systems, you can see a display plane screen which is called HMI, it stands for Human Machine Interface. On this test bench also one basic HMI has been used, in order to display the state of cylinders, valves and all other necessary analogue & digital outputs and inputs to indicate to the operator.

In this thesis HMI design, there are three screens that one of them is the main root screen and two others are sub-screens. The first sub-screen is about all control buttons, state of modes, and indicating the pressure monometers for each chamber. The second sub-screen is only to indicate the Logo of POLITO. One Template screen also has been designed to show the logo of the university in the background of the main root screen. I will explain all about screens and the used icons and symbols on each screen.

Designing the HMI screen is like writing PLC program, it means, first of all, it is necessary to define all the tags which might be needed on all screens of HMI. All components, figures, actuators, and other objects, that want to show or to do something, on the screen must have one HMI tag in the HMI tag table. Obviously, each HMI tag must be attached to one PLC tag or it can be defined as an internal HMI tag. For each of these tags, there is memory storage with the size that depends on the data type of variables.

Two figures below are shown an HMI tag table and an HMI Data block.



Figure

1-8-1- Root Screen Display of HMI

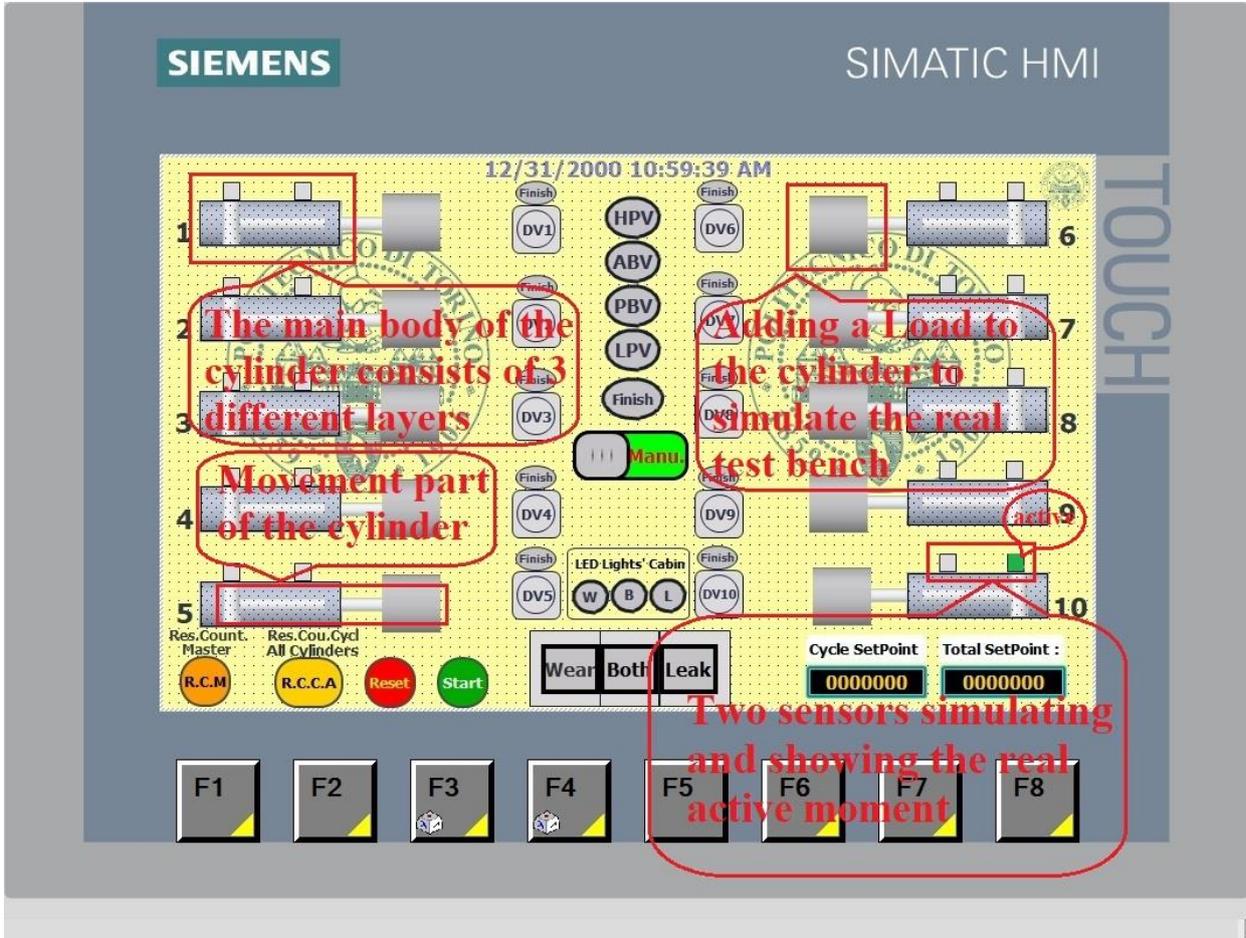
The main root screen is an environment to design all needed objects or templates. On the screen of this HMI, on this thesis, there are many objects consists of cylinders, pushbuttons, light indicators, and counter setpoints.

The first and main items are cylinders

It has three different layers to show the moving piston and the load. One layer is the main body of the cylinder, 2nd layer is the moving part of the cylinder with the load simulation, and the last part is a transparent button that can receive the action from the operator. while the body of the cylinder is touched, it will activate a window Faceplate to show something about exactly that cylinder. By activating the faceplate of each cylinder, all about that cylinder become visible like; some pressure indicators, timers, number of total cycles, number of each cycle are indicating. It is noted that each cylinder can be receptive separately from this faceplate and its own counter can be reset too. For closing and disappearing the faceplate, it is needed to hit the red small square with the X symbol on the left top corner.

Another beautiful part of this animation is the existing two small sensors on the top of each cylinder that is activated while the real sensors are active.

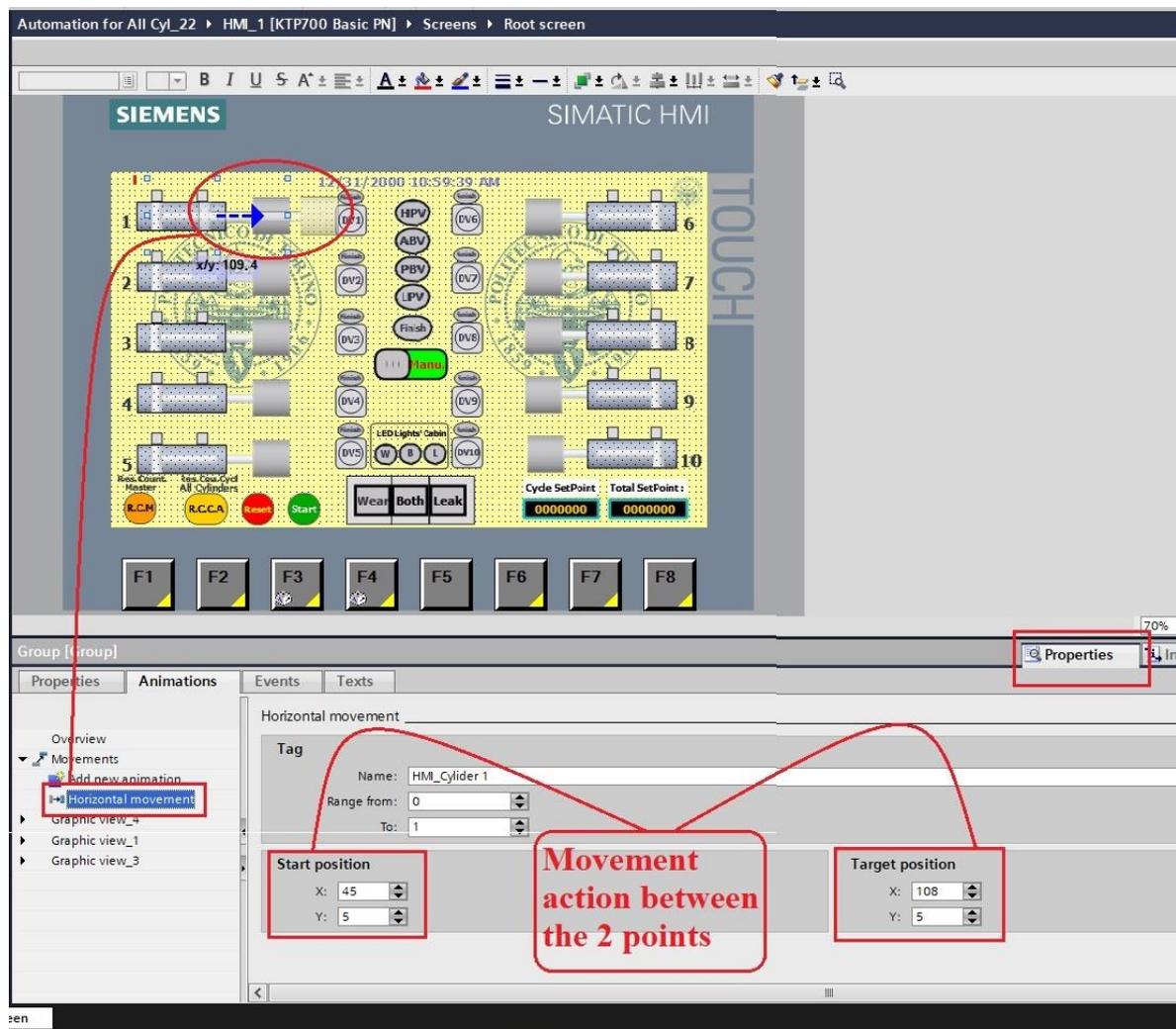
Some brief explanation of the figure is shown below.



Figure

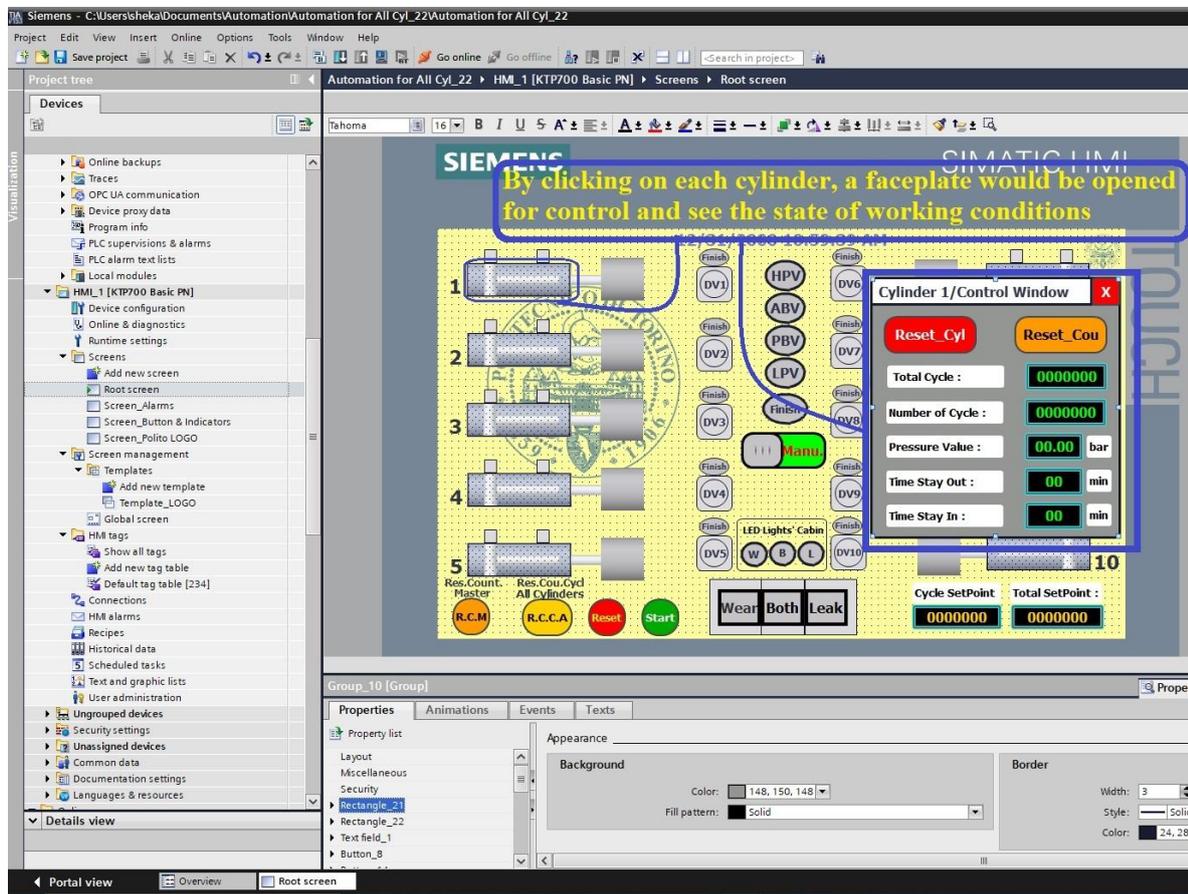
Design and definition of the moving part of the cylinders

The movement should be defined in the animation sub-tab under the properties tab, as it is clear in the figure below. For any movement it is necessary to know the direction of movement, is it horizontal or vertical. In this case, I have chosen the horizontal movement because the cylinders are in this position. Then the starting location with two points X & Y and also the end location with two points X & Y are assigned. All these things should be done in the properties tab → animation sub-tab.



Figure

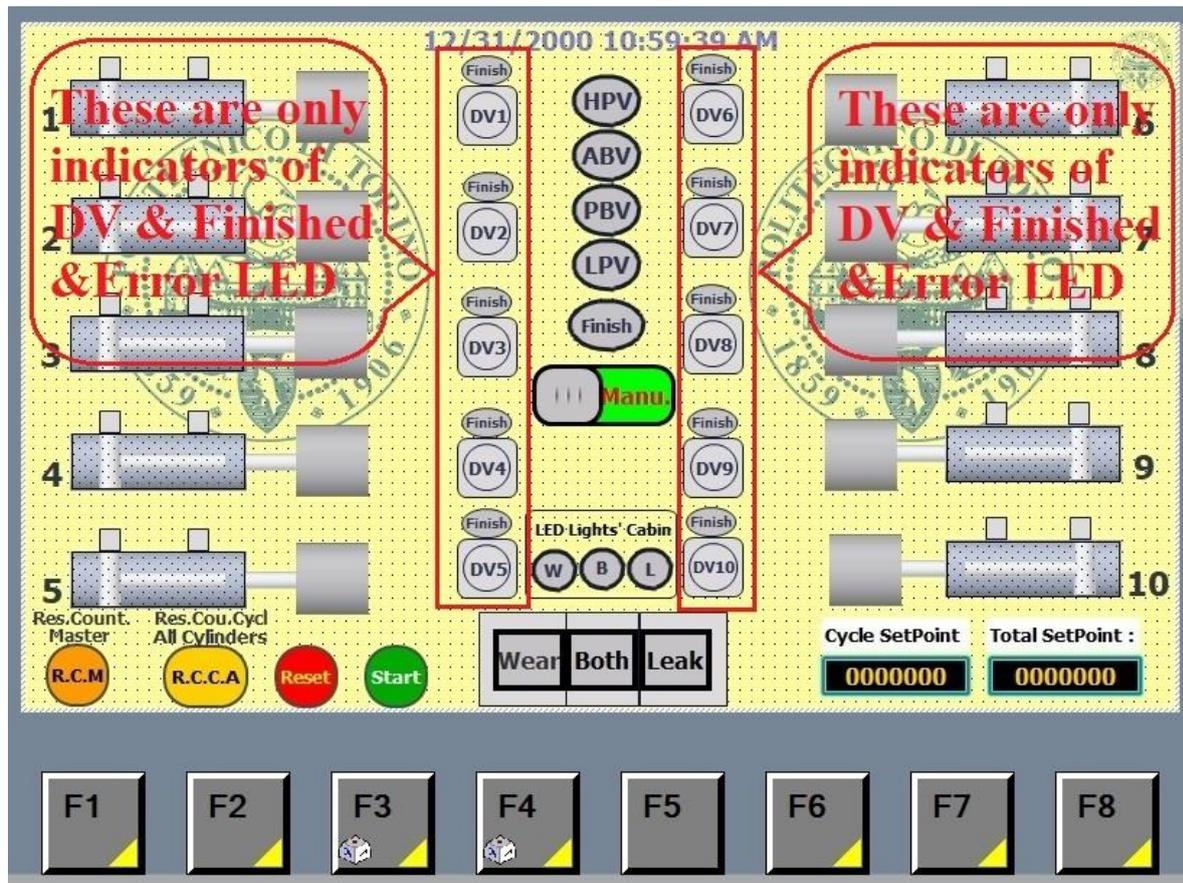
The faceplate is an extra window for each cylinder separately, that is created in the root screen. The status of this window is invisible in runtime screen' but by hitting the body of each cylinder a window faceplate popping up and become visible. It shows some temporary variables like the number of each cycle, the number of the total cycle in WEAR mode test & time stay OUT, time stays IN, in the LEAK mode test. By each faceplate, exactly that cylinder can be controlled to reset it or make zero the number of the counter only for that cylinder.



Figure

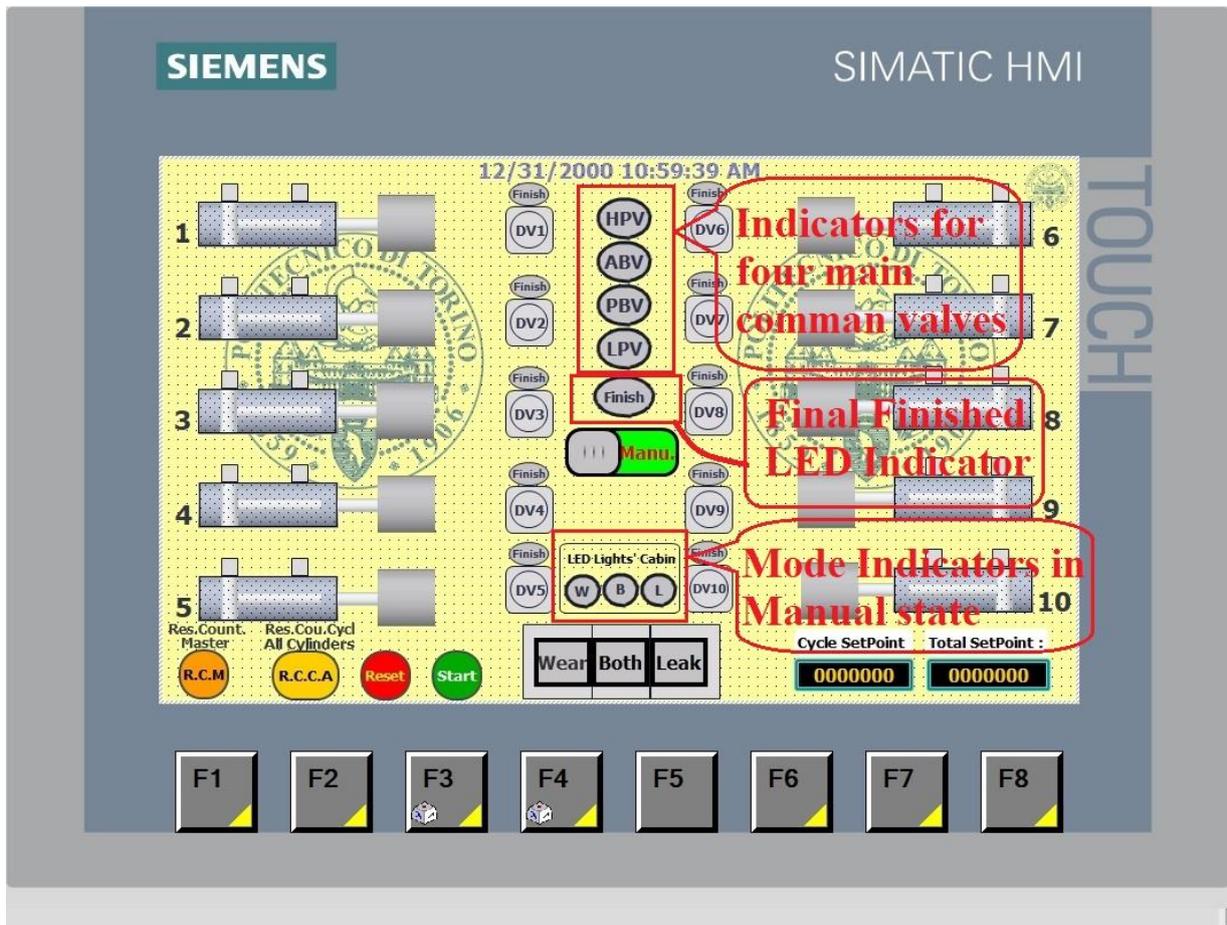
The Second items are LED indicator statuses

These are showing each Distribution Valve (DV), possible Error for each cylinder, and small LED indicator for the Finished cycle of each cylinder. It is shown in the figure below.



Figure

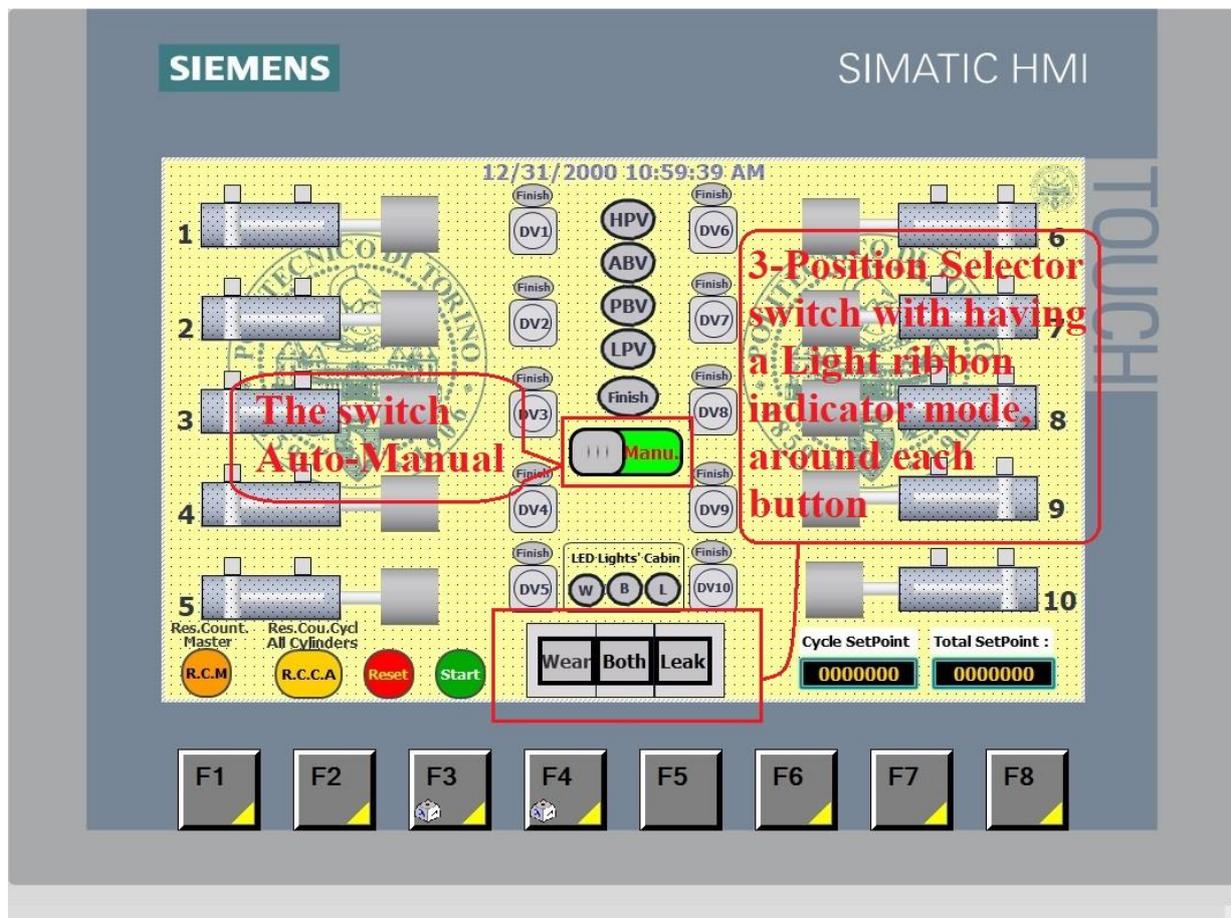
The rest of the LED indicators which are about the Final finished LED, LED mode indicator for manual status, and four common valve states. Four common valves are High-Pressure Valve, Low-Pressure Valve, Anterior Block Valve, and Posterior Block Valve. All about these are shown in the figure below.



Figure

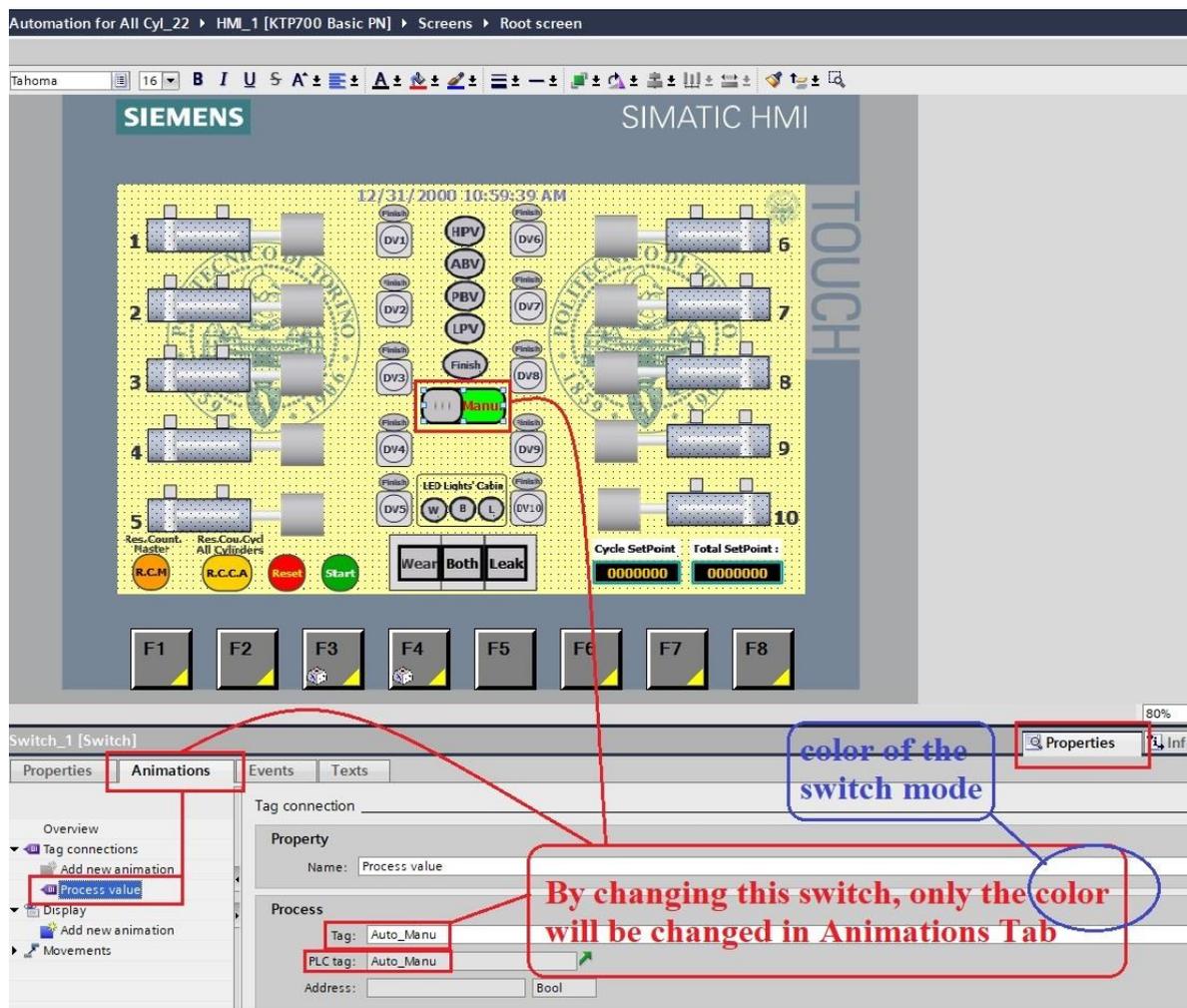
The 3rd part of Root Screen is Auto-Manual and Selector Mode Test

The main switch that allows the test bench to be run by HMI, is **Auto-Manual Switch**, which is located exactly in the middle of the Display Screen. By putting this switch in Auto mode status, the test bench can be run by selecting one of the modes only from HMI panel. For any changes in mode selection, 1st it needs to be reset and then choose another mode. When this switch is in Manual mode status, the selector switch on HMI does not work, only the intended mode can be selected from the physical electric switchgear panel. The Selector Mode Switch has three positions for three different modes of the test bench. Behind this switch, there are some rebounds which are light indicators for that mode. This switch can work only when the Auto-Manual Switch is in the Auto Mode position.



Figure

The definition of colour changing of the Auto-Manual button on HMI has been shown in the figure below. By selecting the switch on the HMI, the window below screen will appear, then from the properties tab go to animation sub-tab, then the related tag from the PLC tag table must be attached. The functionality of this switch can be set in the next tab which is called “Event”. it is exactly as same as the animation set. These tags must already be defined in the PLC tag table.



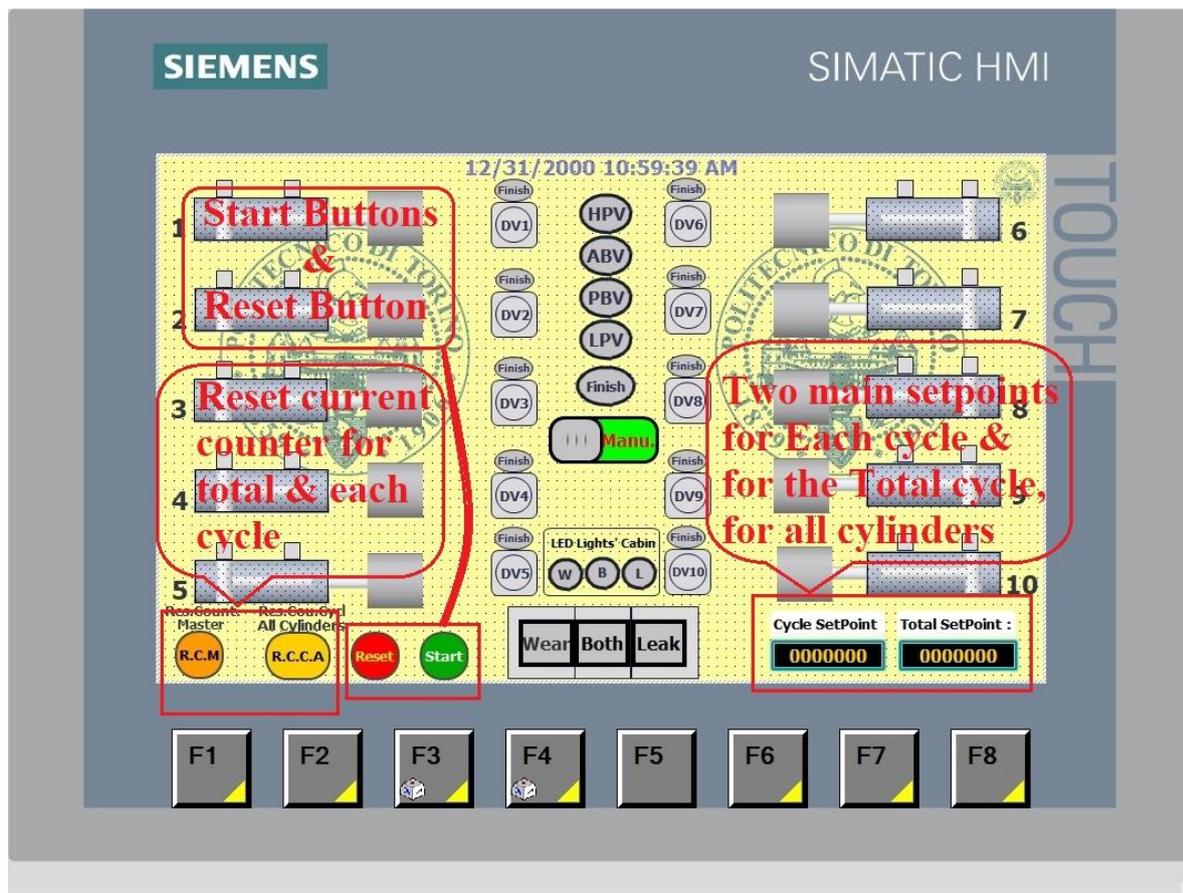
Figure

The 4th part of Root Screen is about Start, Reset, Counters & Setpoints

There are two Start & Reset push buttons on the HMI screen to control the test bench from a remote area.

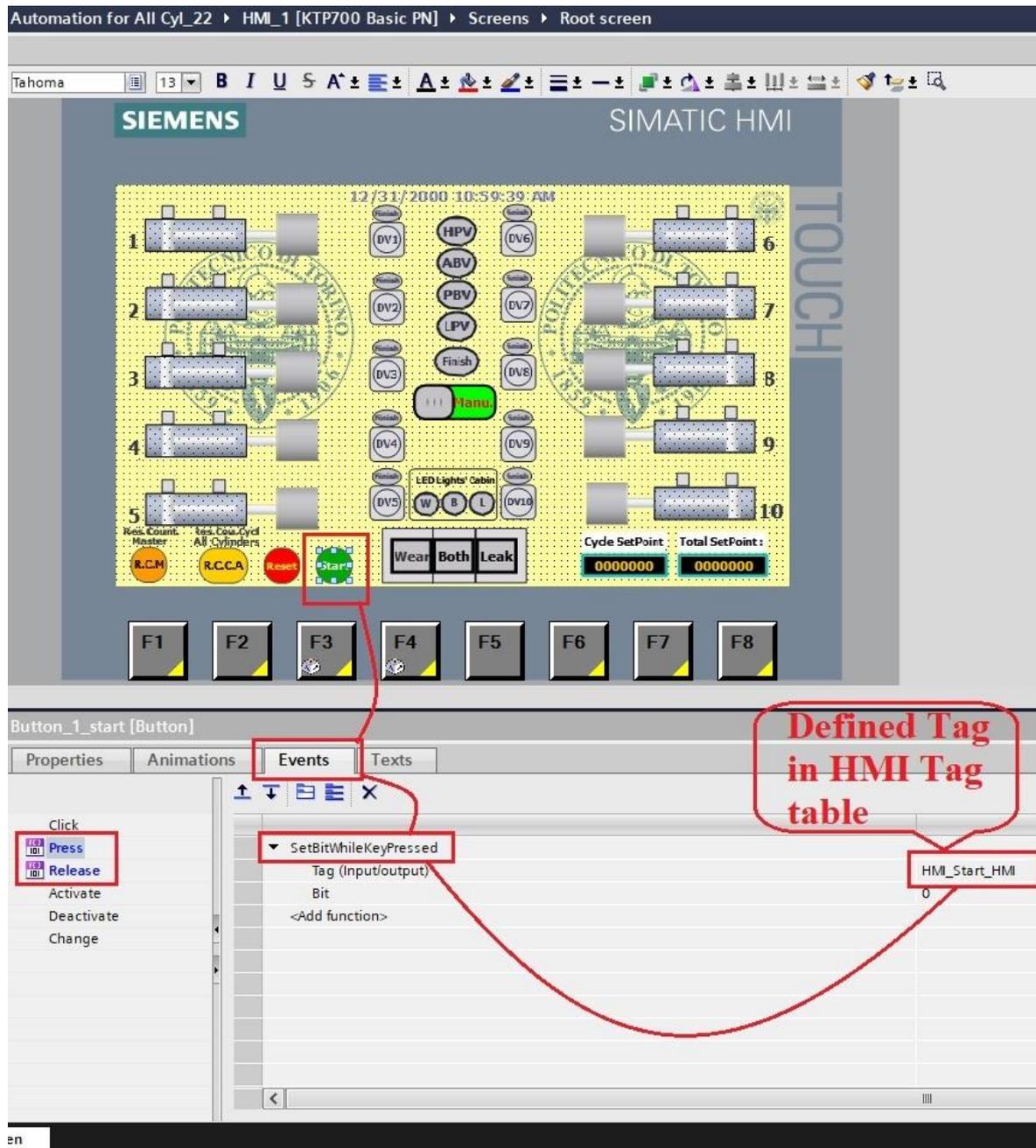
Besides, there are two pushbuttons for resetting the counters, one of them reset the “total number” of the counter, and the next one reset the counter of “each cylinder” for all cylinders, in the same time.

Also, there are two small boxes for entering the setpoint number, one of them is for “each cycle” and the other one is for “total cycle”. These setpoints are related to all cylinders.



Figure

The definition of the Start button on the HMI screen should be done in the properties tab, then go to sub-tab that is called “Event” then define the two states of the Start button. This setting for all buttons like Reset cycle or Reset counters which receive an action, it is the same for defining and assigning a tag for them.

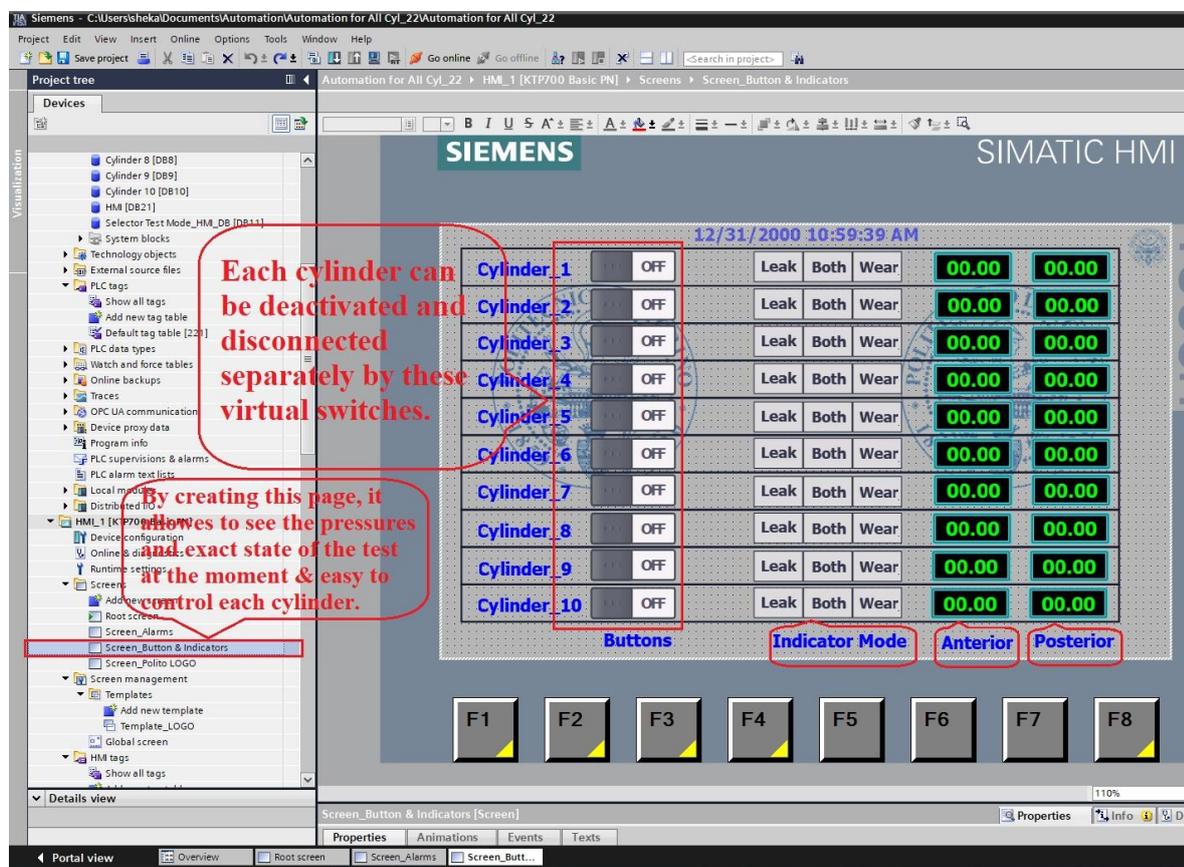


Figure

1-8-2- Button & Indicators Screen

The main goal of creating this screen is to activate and deactivating each cylinder separately. Besides another aim of this is to indicate the air pressure of each chamber for all cylinders, while they are under the LEAK test. In addition, it is important to see the exact model of the test bench while it is working in “Both Modes Test”, which is shown by 3 lights for each cylinder.

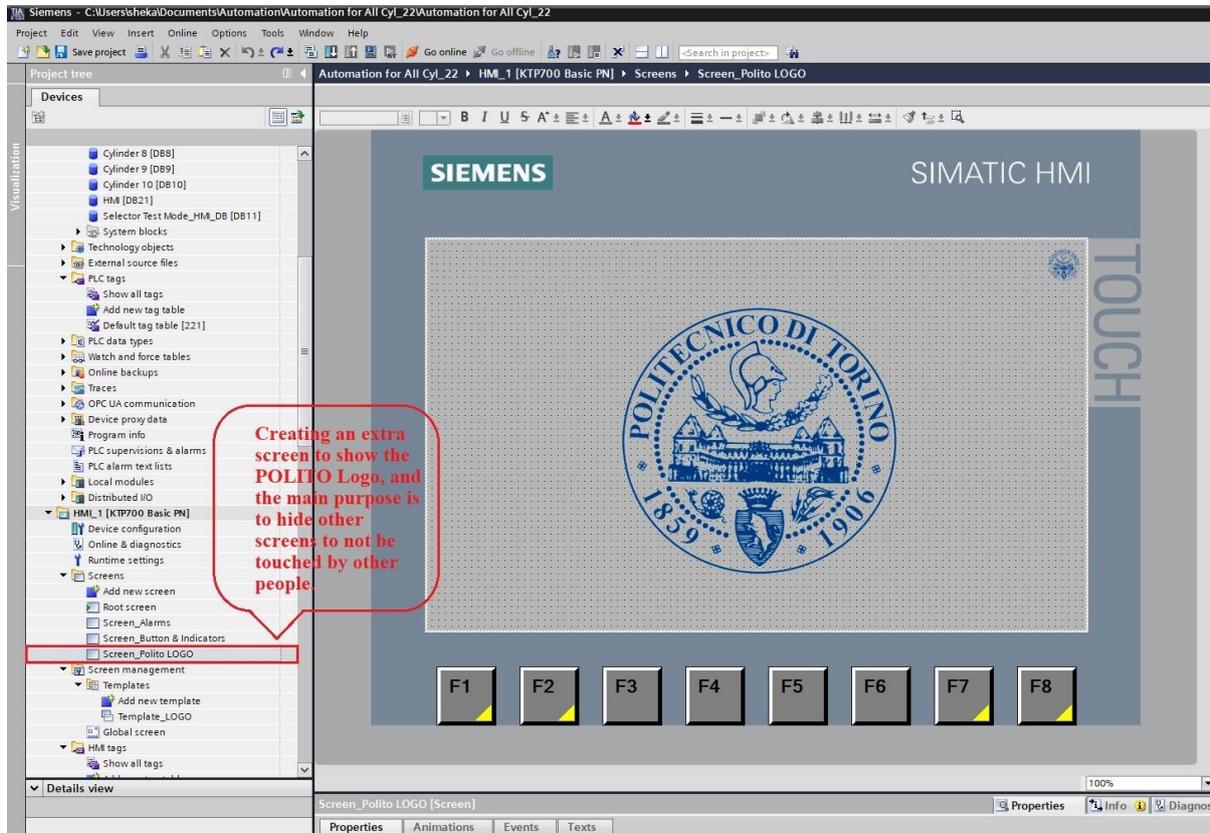
In the photo below, all these things have been clarified and explained.



Figure

1-8-3- POLITO Logo Screen

The target of this screen shows the Logo of the University POLITO and more important is while we do not want anybody to touch the HMI screen, we can activate this screen by pushing one of the physical Functional buttons which are located in the bottom of the HMI (F7).

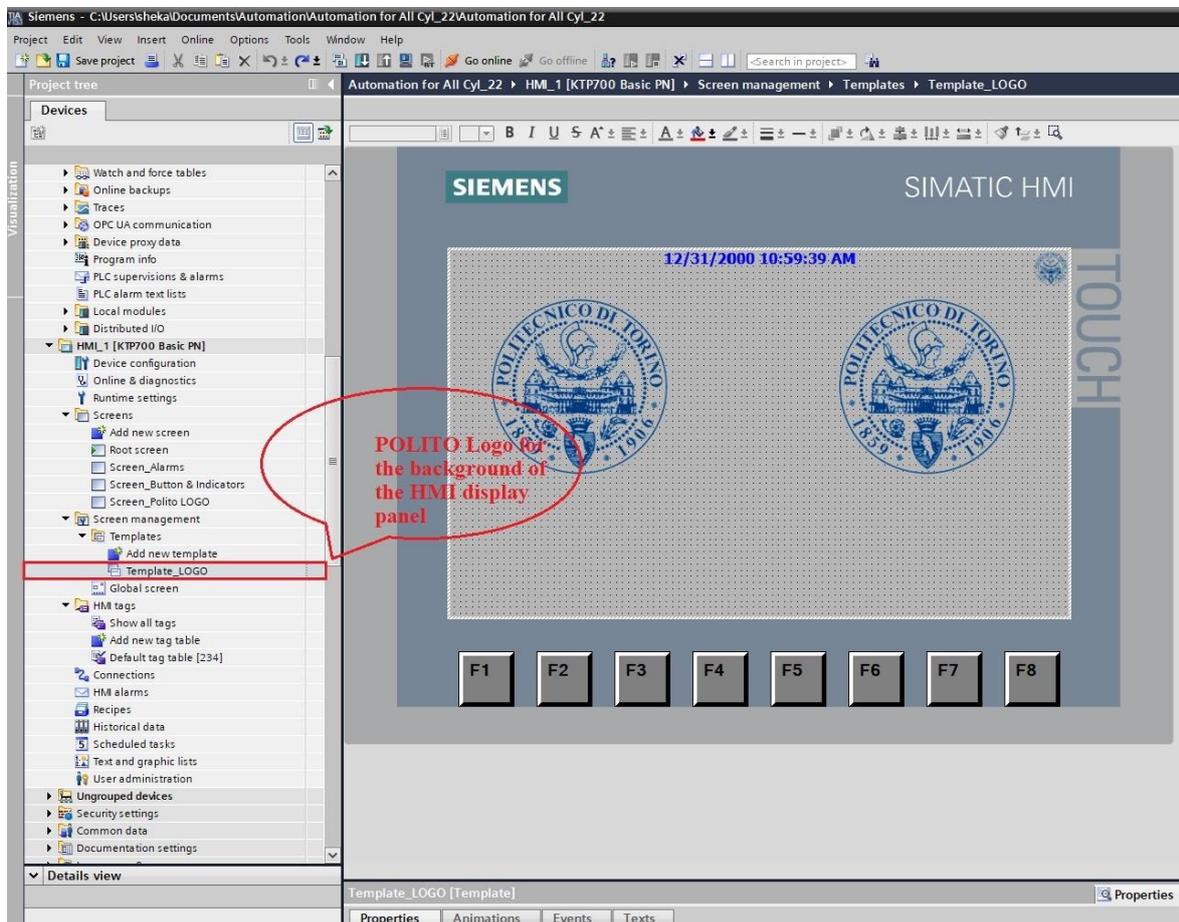


Figure

1-8-4- POLITO Template Screen

While the main root screen is in run mode on HMI, it is to show, that the project has been designed and performed by a student of the Polytechnic University of Turin.

This page can be designed in the template sub-folder under the folder of screen management in the HMI project.



Figure

C. 1-9- Analogue Data Saving in HMI Storage

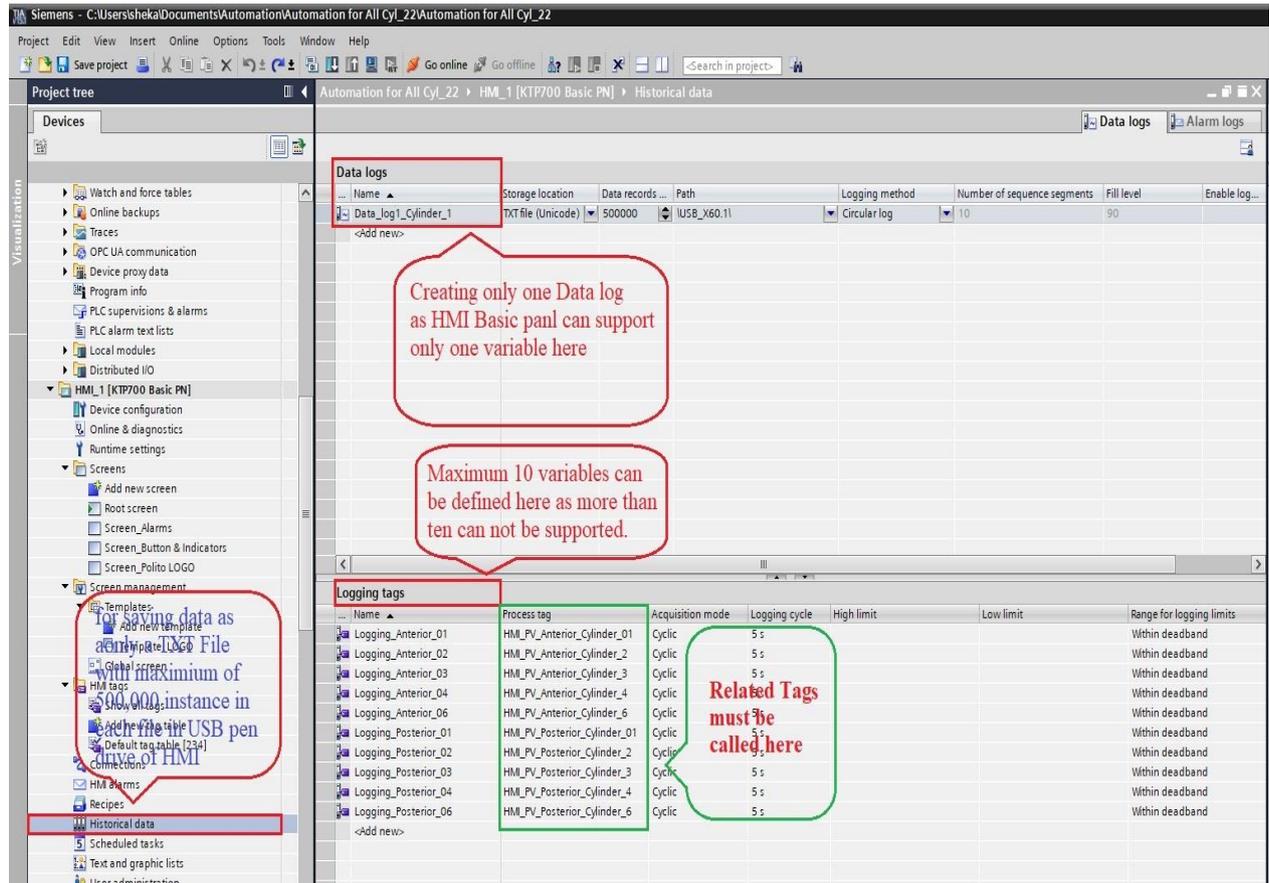
In many automation systems, it is necessary to save data and create a text file or Excel file to have the data for analyzing variables. It allows us to analyze and assess the system behaviour.

So, while there is an HMI (Humane Machine Interface) or SCADA (Supervisory Control and Data Acquisition) in the project, it would be easy to save some variables as a TXT file or Excel file in the storage of HMI panel, which is a USB2 pen drive storage. It depends on the type of HMI that allows how many variables can be saved in the HMI storage. It is noted that with basic HMI panel only one “data log” and for each “data log” only 10 variable “lugging tags” can be defined.

In this case, some variables for each cylinder have already assigned and defined. So, while they can be saved in HMI storage by going to the “Historical Data” sub-menu of the HMI project menu, it can be done easily.

When you are on the page of “historical data” there are 2 different windows upper one which is called “data logs” and downer one which is called “lagging tags”. In the upper window we need to assign a “data log” of the project and in each one you should define the storage location, the number of data records, the directory of saving data, logging method, number of sequence segments, fill level, and also you can enable lugging at runtime started. For each defined data log in the “data logs” window, you can define some logging tags in the downer window. For defining each lagging tag, you must define “process tag” from one of tag tables, acquisition mode, and lunging cycle, also you have the possibility to define the high limit and the low limit range of saving data at the moment. After defining these variables, in the HMI storage, in a folder, a text file would be created with saving 10 variables inside a TXT file. it is very important to know dance the maximum number of data records is 500,000 with this PLC and HMI device.

However, there are some other ways of saving data separately in a different text file. and this part of the project needs to be done by one of the other students after me. Some points are shown in the figure below.



Figure

1- Chapter 2: Electrical and hardware implementation

2-1- Physical Utilized components in this test bench

In any project, it is necessary to estimate and evaluate the needed suitable components. For each component, there are a variety of kinds that it is necessary to be careful to choose the right one with the needed characteristics. In this project there are some automatic, electrical, electromechanical, pneumatical and mechanical components, I have used for this test bench. In this part, I will try to show and explain them in detail to more illustrate the project.

➤ Utilized components in the project consist of;

1. PLC device S7-1512



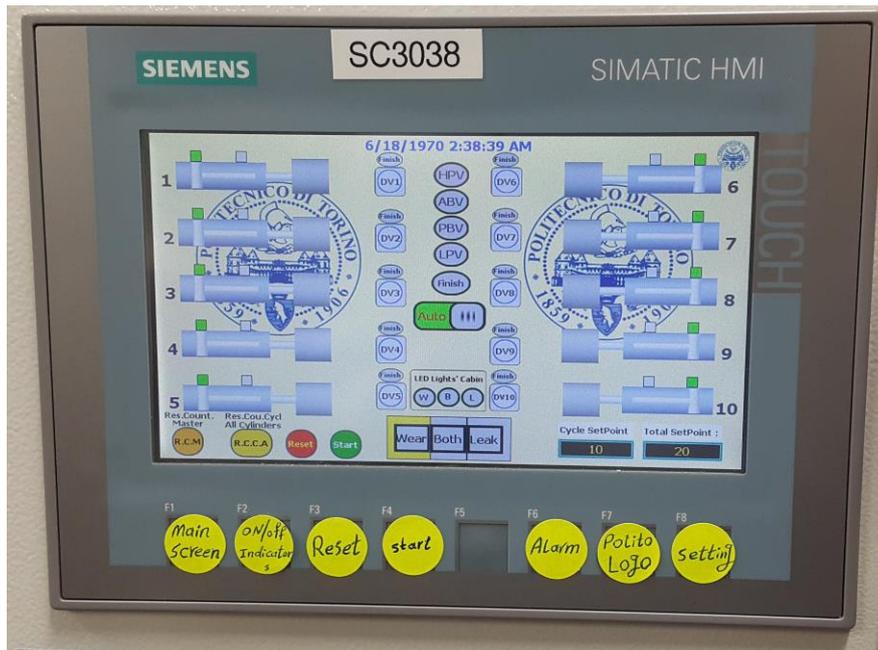
Figure

2. Remote IO controller ET200

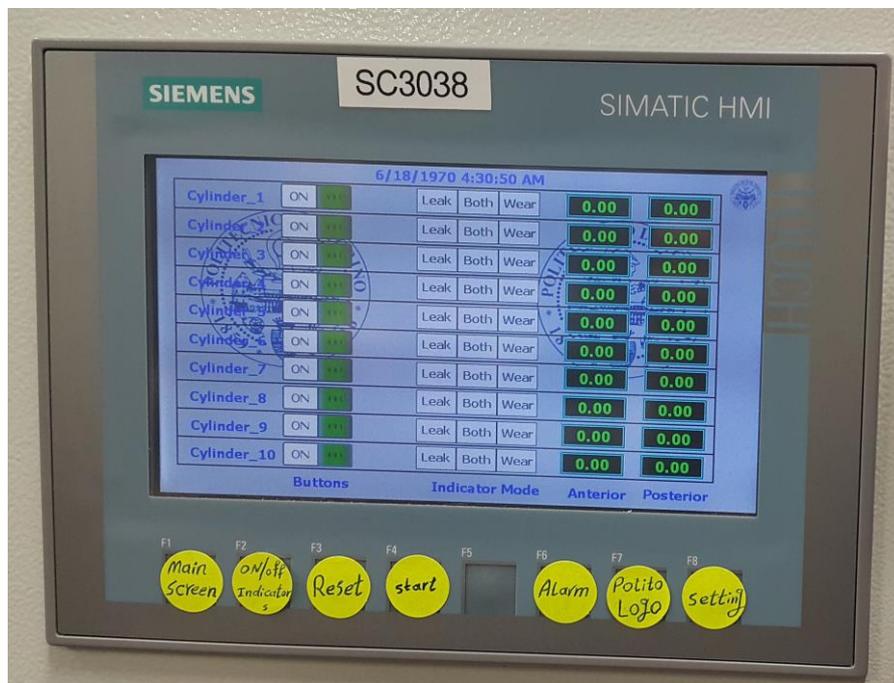


Figure

3. HMI display Basic panel



Figure



Figure

4. Scalance Switch connection



Figure



Figure

5. PROFINET cable connection



Figure

6. Counter cycle



Figure

7. Electric switchgear



Figure

8. LED signal light indicators (In Figure below)
9. Open contact Start push button (In Figure below)
10. Close contact Reset push button (In Figure below)
11. Close contact Emergency push button (In Figure below)
12. Selector mode switch (3-position switch) → (In Figure below)



Figure

13. Close contact Door-Microswitch
14. Plastic pipe
15. Branch pipe connector
16. Ten-kilo metal Load
17. Metal protection guard of whole test bench
18. Electric breaker
19. Reed sensor
20. Copper Wire
21. Sar sim
22. Omega rail
23. A 24V-Electric power supply



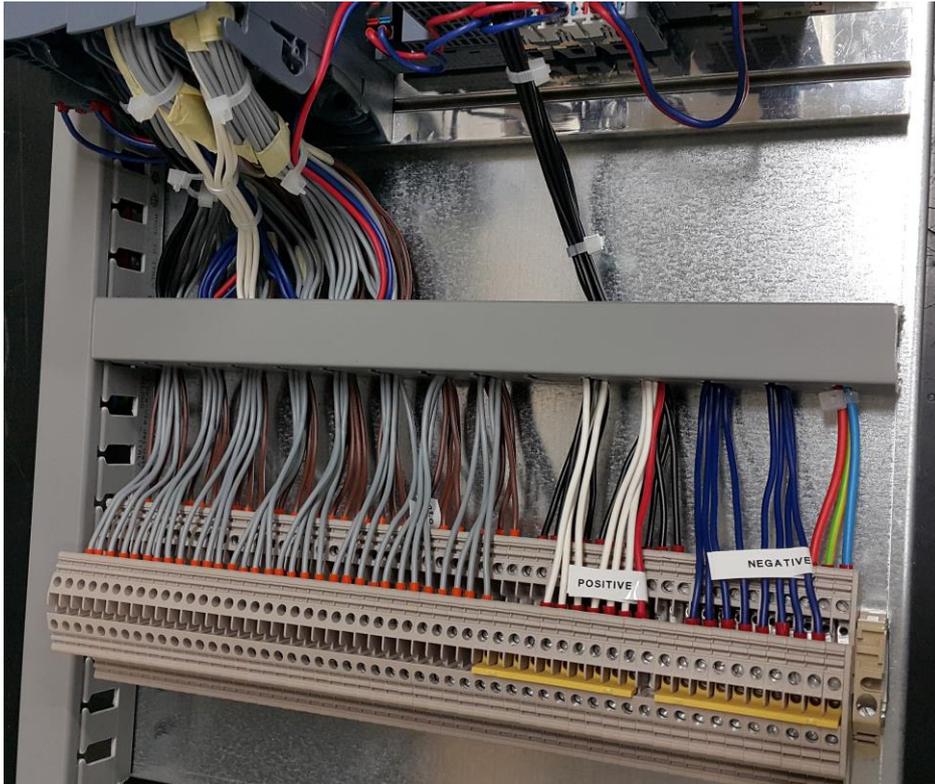
Figure

24. Plastic belt (In Figure below)

25. The comb of terminal jumper (In Figure below)

26. Plastic channel (In Figure below)

27. Electric interconnection terminals (In Figure below)



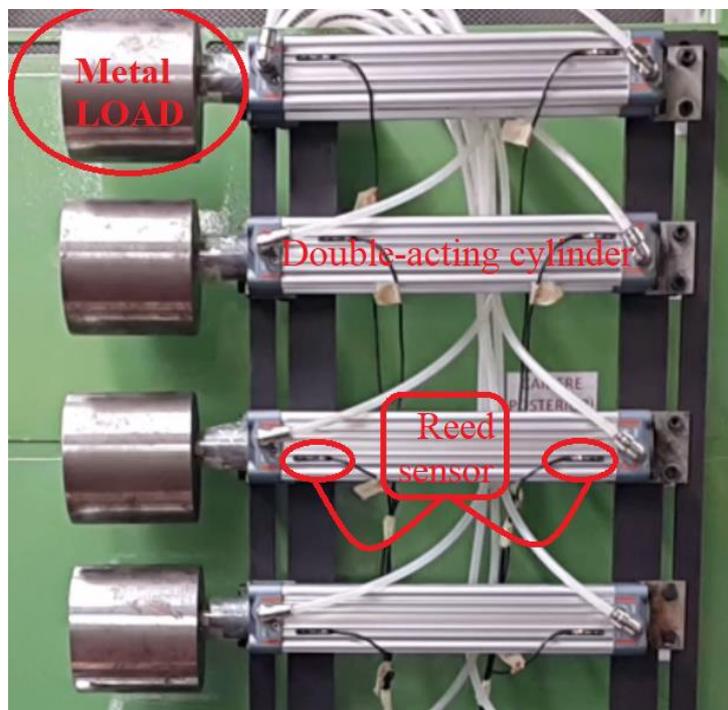
Figure

28. Analog pressure sensors



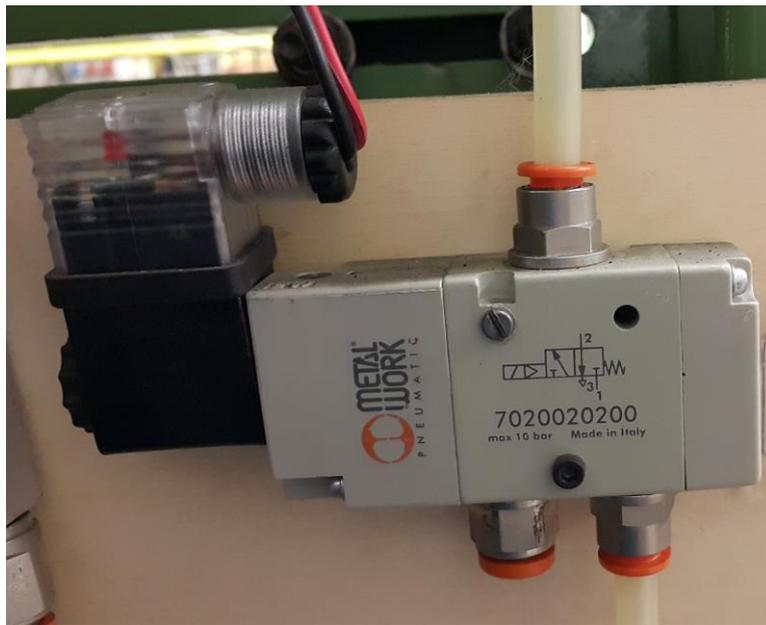
Figure

29. Double-acting cylinders



Figure

30. Electric monostable valves 3/2



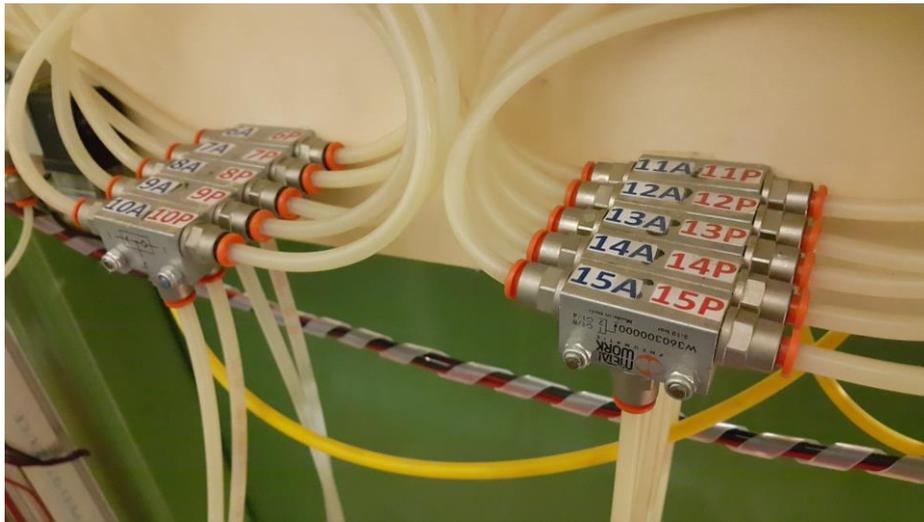
Figure

31. Compact 10-electric monostable valve 5/2



Figure

32. Pneumatic OR gates



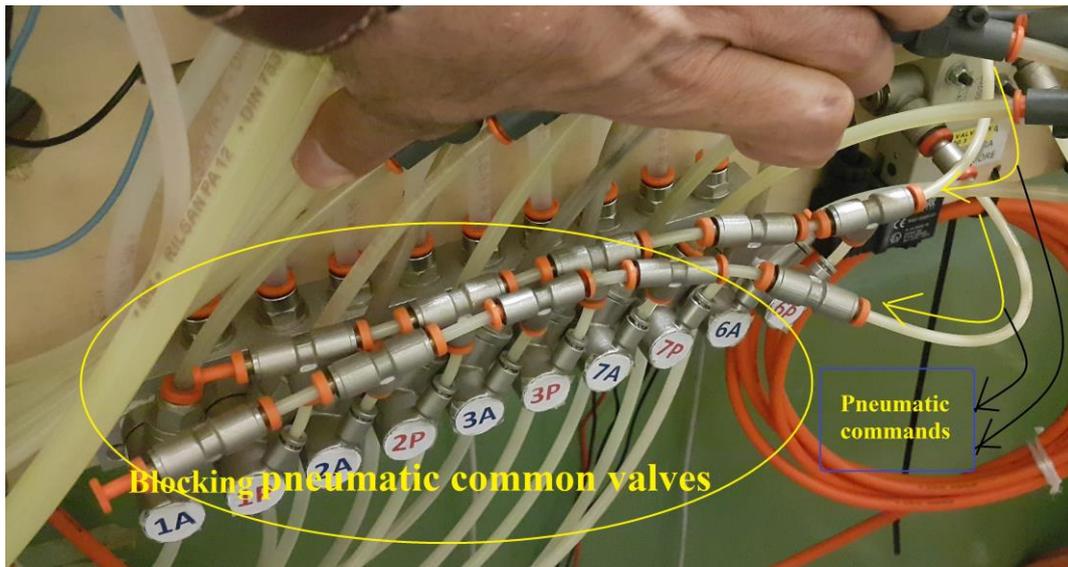
Figure

33. Electro-pneumatic monostable valves



Figure

34. Blocking pneumatic valves



Figure

35. Pressure regulators

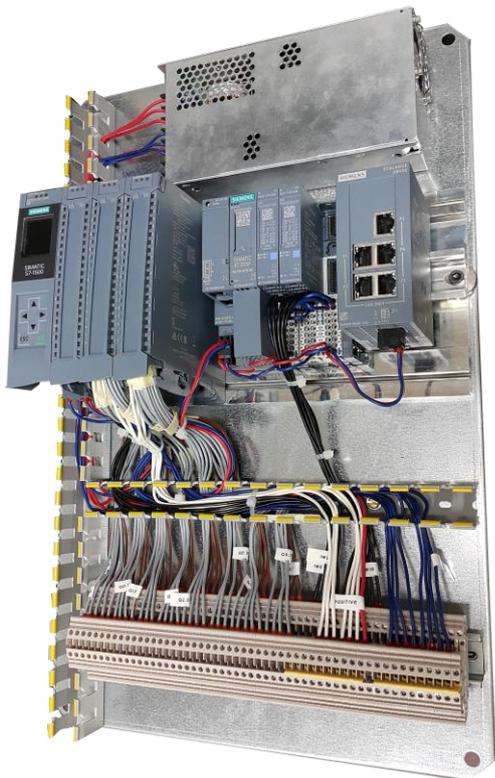
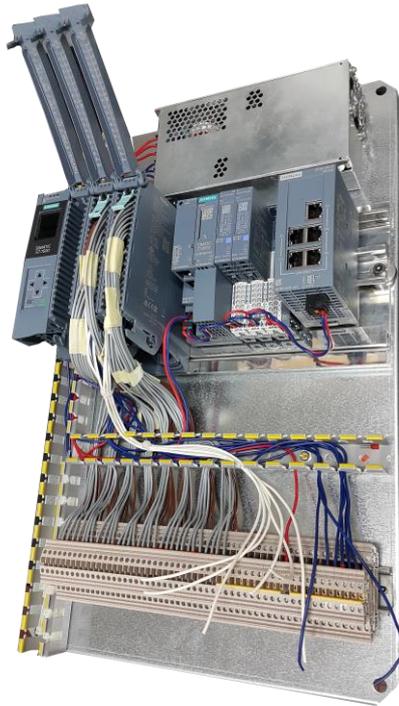


Figure

2-2- The implementation & Assembling of the project



Figure



Figures

Conclusion and Suggestion

A brief overview of all the projects is found in this chapter. this project consists of some different parts.

The first part is the connection and installation of hardware and mechanical structure of cylinders, valves, and pneumatic components on the test bench.

The second part of the project is preparing, assembling, implementing electrical and automation components inside the electric switchgear correctly and suitably, and then installation of electric switchgear on the test bench in a suitable place.

The third part is a significant section of the project which is, writing and designing automation and PLC program to control the cylinders in two different separated modes WEAR and LEAK, also in performing two tests in the same mode, which is called Both Modes, successively and continuously.

The Forth part is matching the software and hardware program which is a visible part of the project. In the last section, there might appear many small problems and incompatibilities. So good troubleshooting is required to find the incompatible parts and solve the problem, which is usually related to sensors, connections and sometimes wires.

Future developments and suggestions

In any system future development would happen due to rapidly progressing the technologies. For this test bench, I can give some recommendations to improve it in the upcoming days.

In the project of HMI, it is possible to define some errors by popping up some notifications while the wrong button is pressed or while some setting is needed.

In any case, someone else might write another program that has fewer program lines or fewer blocks with the same functionality, however, I tried to do my best.

All the programming codes in ladder Language is available at the end of this thesis.

References

Automating with SIMATIC S7-1500_Configuring, Programming and Testing with STEP 7 Professional **by Hans Berger's book**

- Industrial Automation_Hands-On_by **Frank Lamb**_ Copyright © 2013 by McGraw-Hill Education
- SIMATIC S7-1500 Industrial Software_SIMATIC Safety - Configuring and Programming_**Programming and Operating Manual**
- Manual Function of PLCSim Advanced;

https://cache.industry.siemens.com/dl/files/153/109739153/att_895955/v1/s7-plcsim_advanced_function_manual_en-US_en-US.pdf

- Drives and Control for Industrial Automation **by Tan Kok Kiong, Ph.D. & Andi Sudjana Putra**
- Handbook of Industrial Automation edited **by Richard L. Shell_Ernest L. Hall_University of Cincinnati_Cincinnati, Ohio**
- SIEMENS SIMATIC S7 PROFINET mit STEP 7 v15_ **Function Manual**
- SIEMENS SIMATIC S7-1500, ET 200SP, ET 200pro_Structure and Use of the CPU Memory_**Function Manual**
- TIA Portal Module 042-201_WinCC Advanced with TP700 Comfort and SIMATIC S7-1500

Appendix and the Main complete PLC Program and Codes

The last section of the thesis is the main Programming code which is obviously the most important part of this assignment. As it has been required to be put at the end of the thesis, it has attached in the following pages. However, some screenshots of the main and important networks of the program have been attached in the exact part of its own explanations among the thesis illustration.

Once again, Many Thanks to the Polytechnic University of Turin in Italy and all professors who are attempting to flourish the talent of new students especially foreigners.

Many Thanks

God

Automation for All Cyl_22 / PLC_6 [CPU 1512C-1 PN] / Program blocks

Main [OB1]

Main Properties

General

Name	Main	Number	1	Type	OB
Language	LAD	Numbering	Automatic		

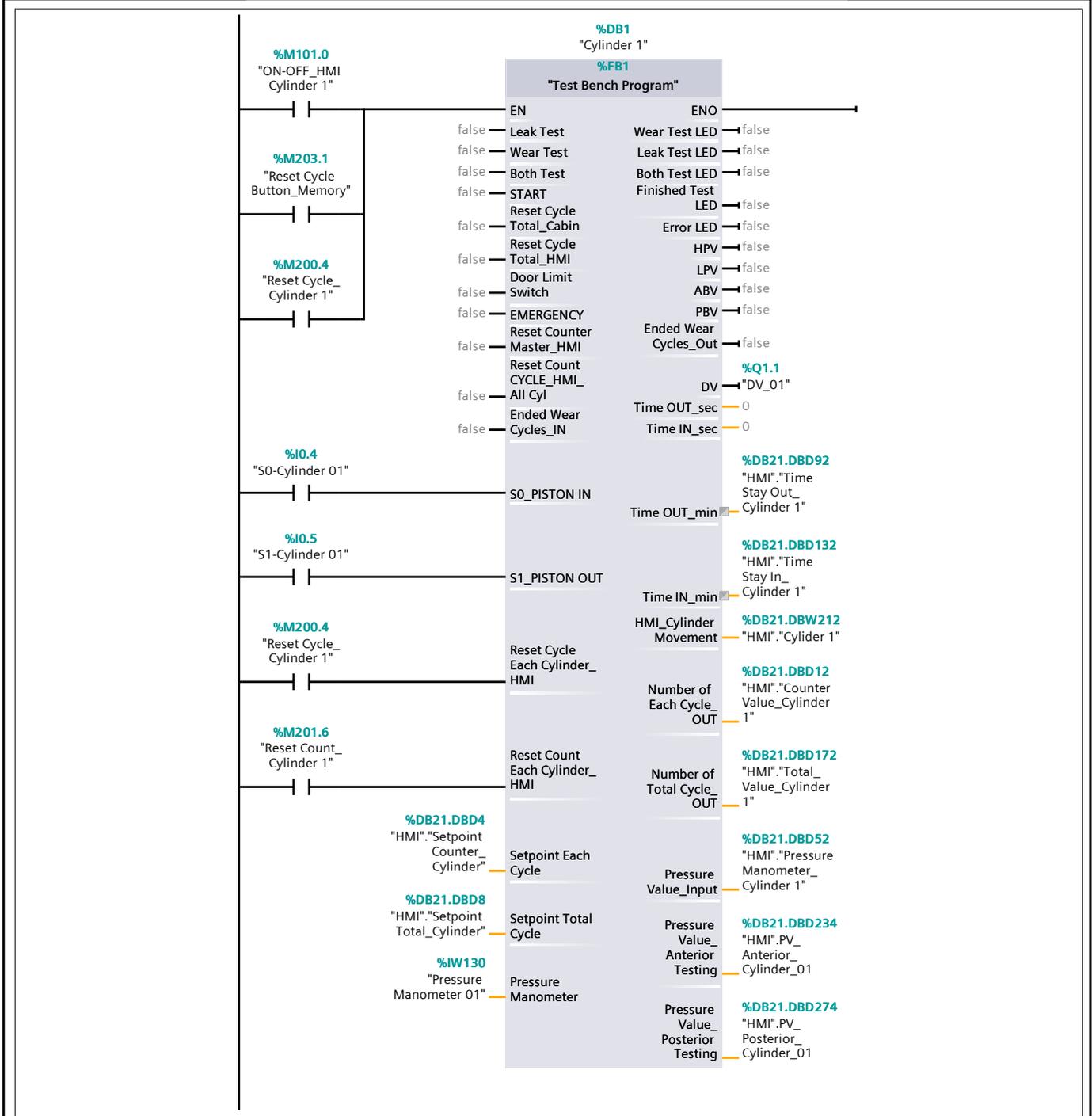
Information

Title	Main Automation WEAR & LEAK Prog. for ALL Cylinders" Designed & Implemented by MOHAMMAD SHEKASTEH	Author		Comment	All the Rights of this Program is reserved for MOHAMMAD SHEKASTEH
Family		Version	0.1	User-defined ID	

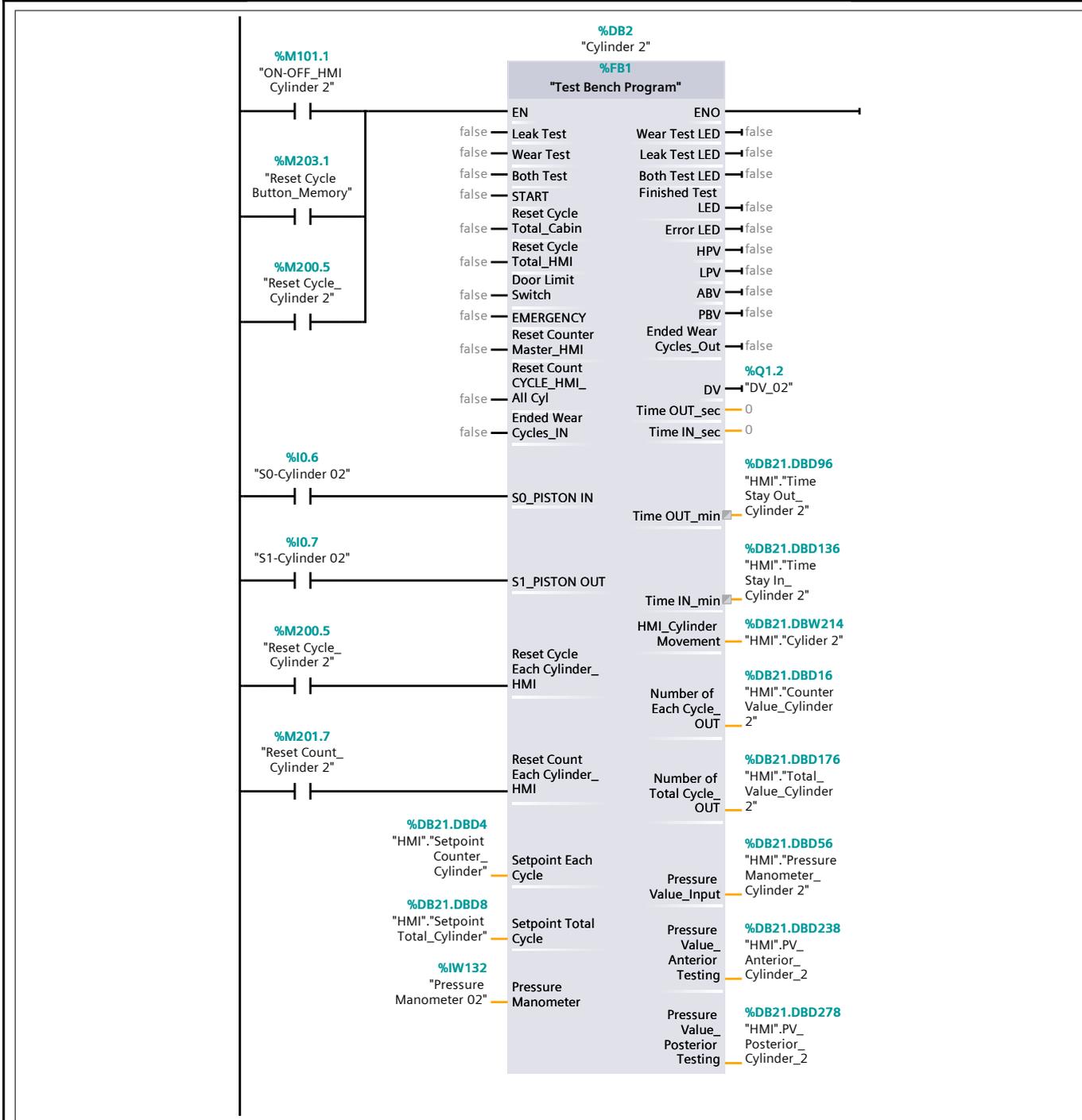
Main

Name	Data type	Default value	Comment
▼ Input			
Initial_Call	Bool		Initial call of this OB
Remanence	Bool		=True, if remanent data are available
Temp			
Constant			

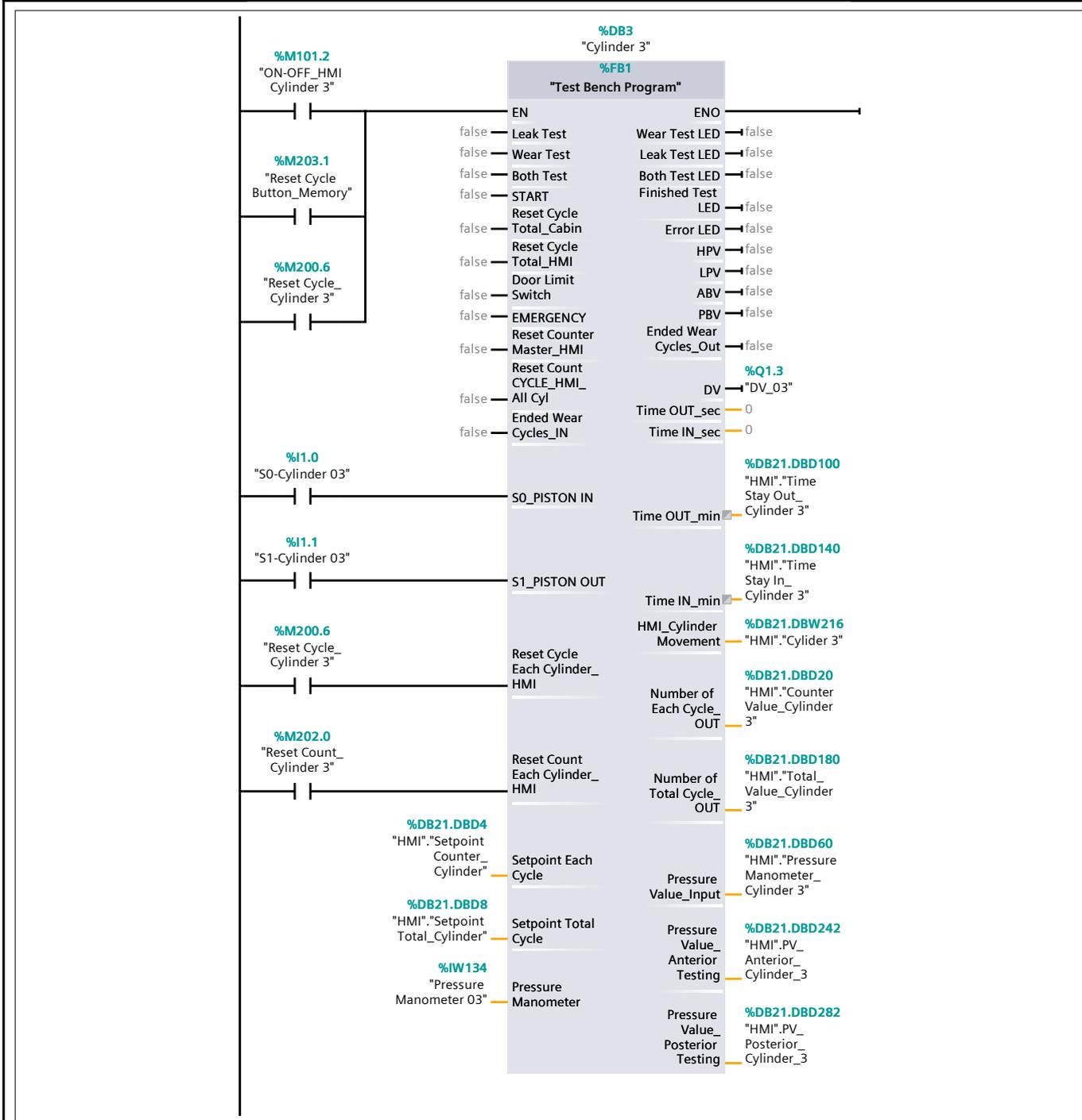
Network 1: Cylinder 1



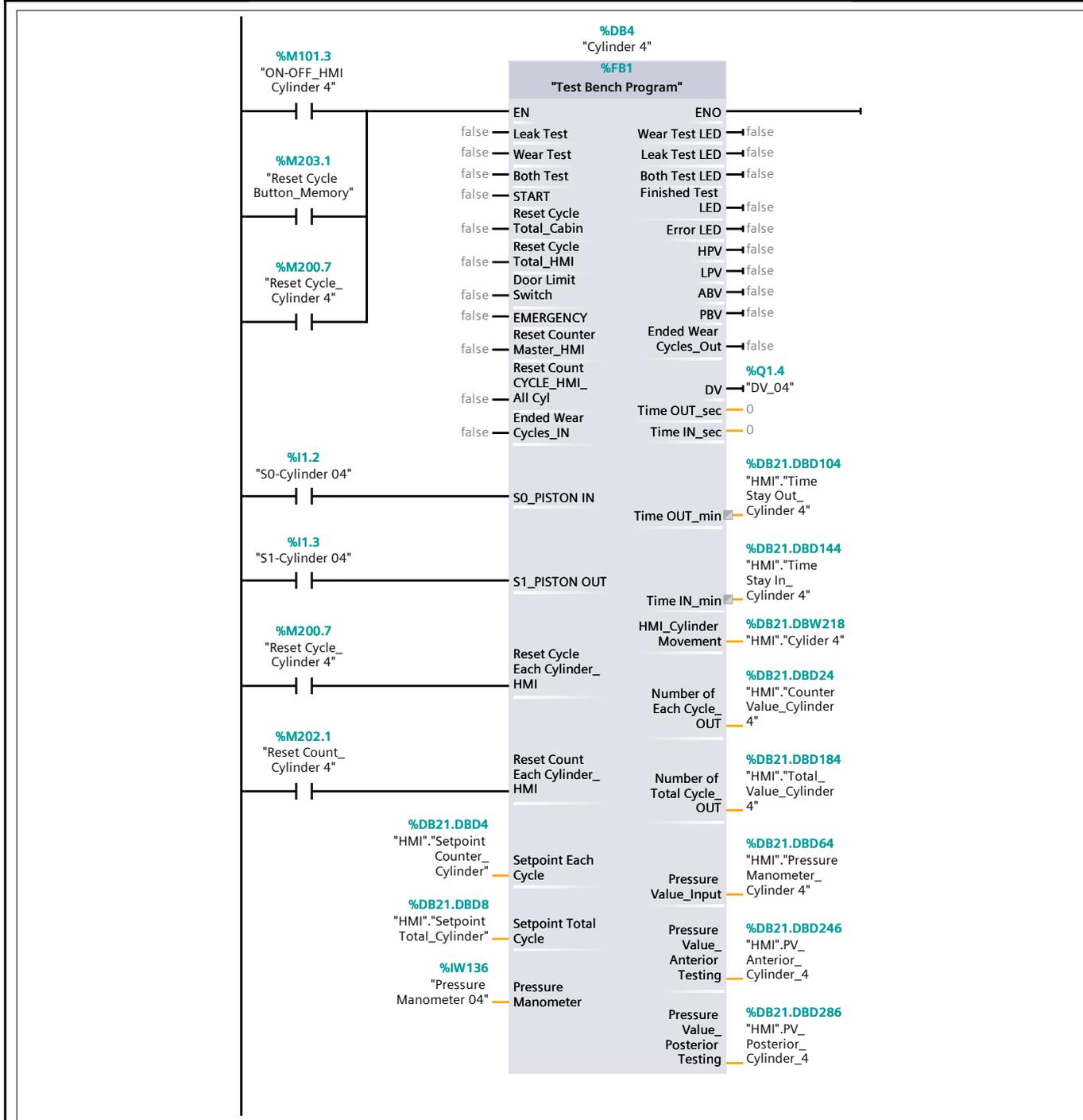
Network 2: Cylinder 2



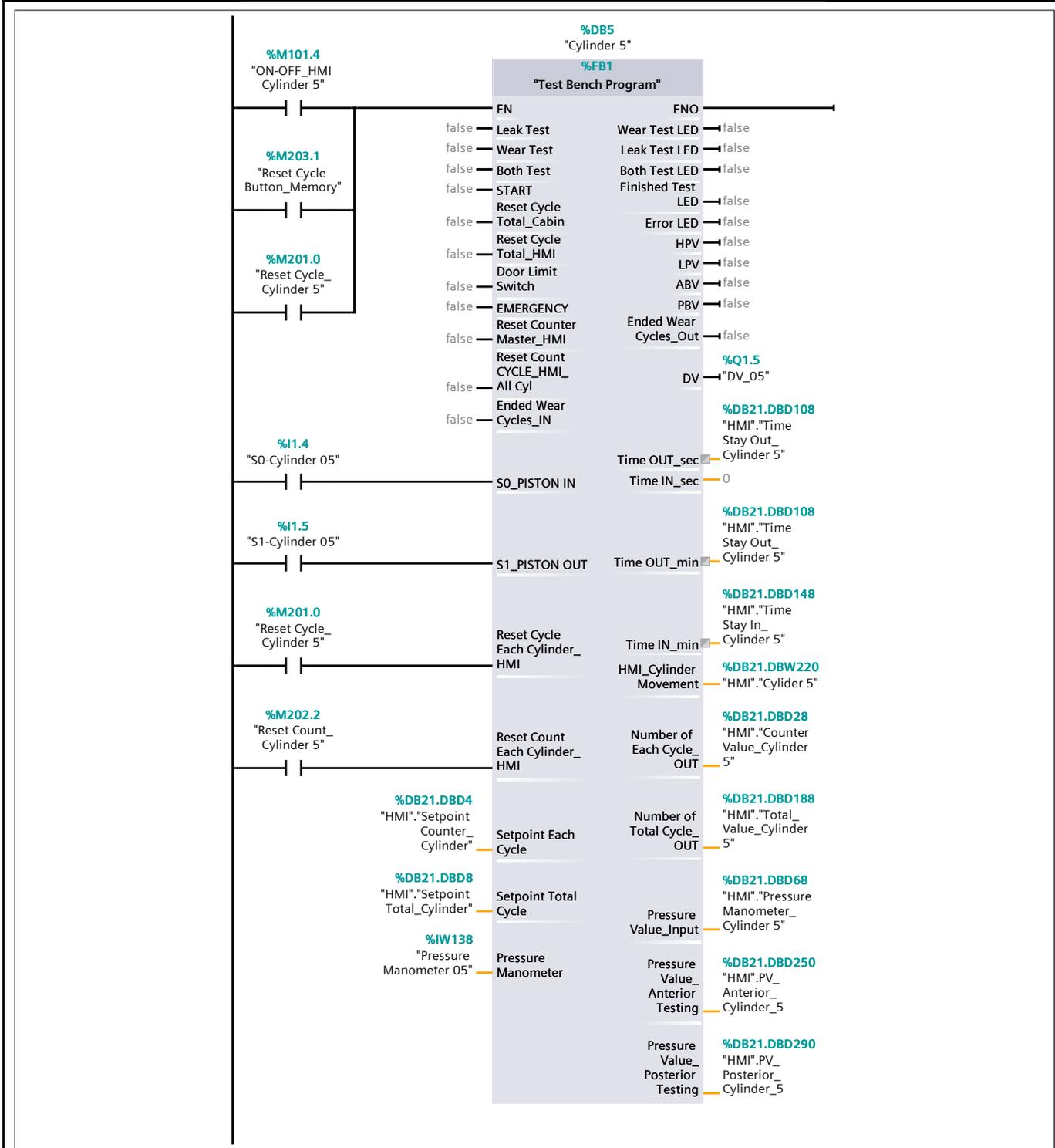
Network 3: Cylinder 3



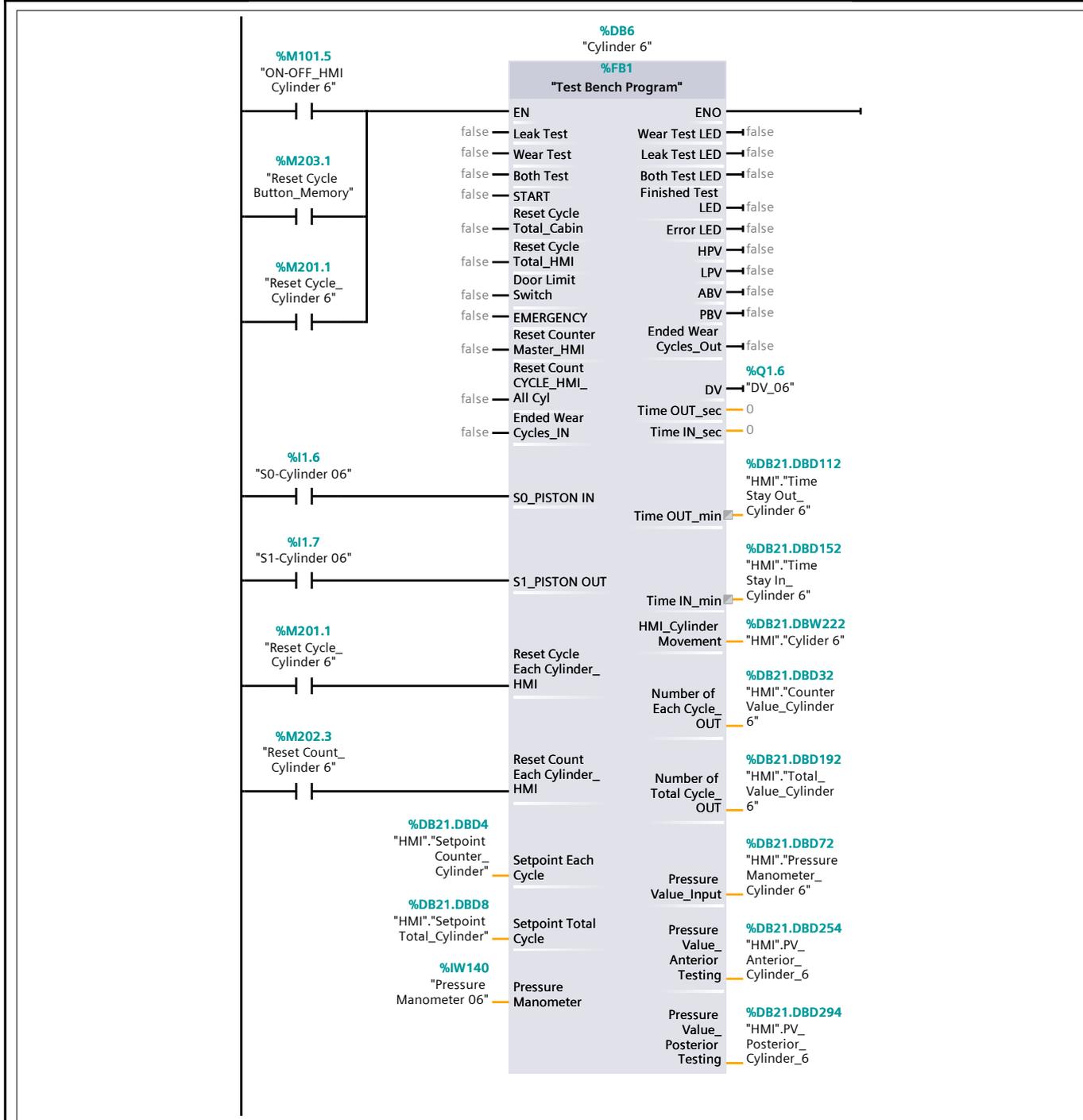
Network 4: Cylinder 4



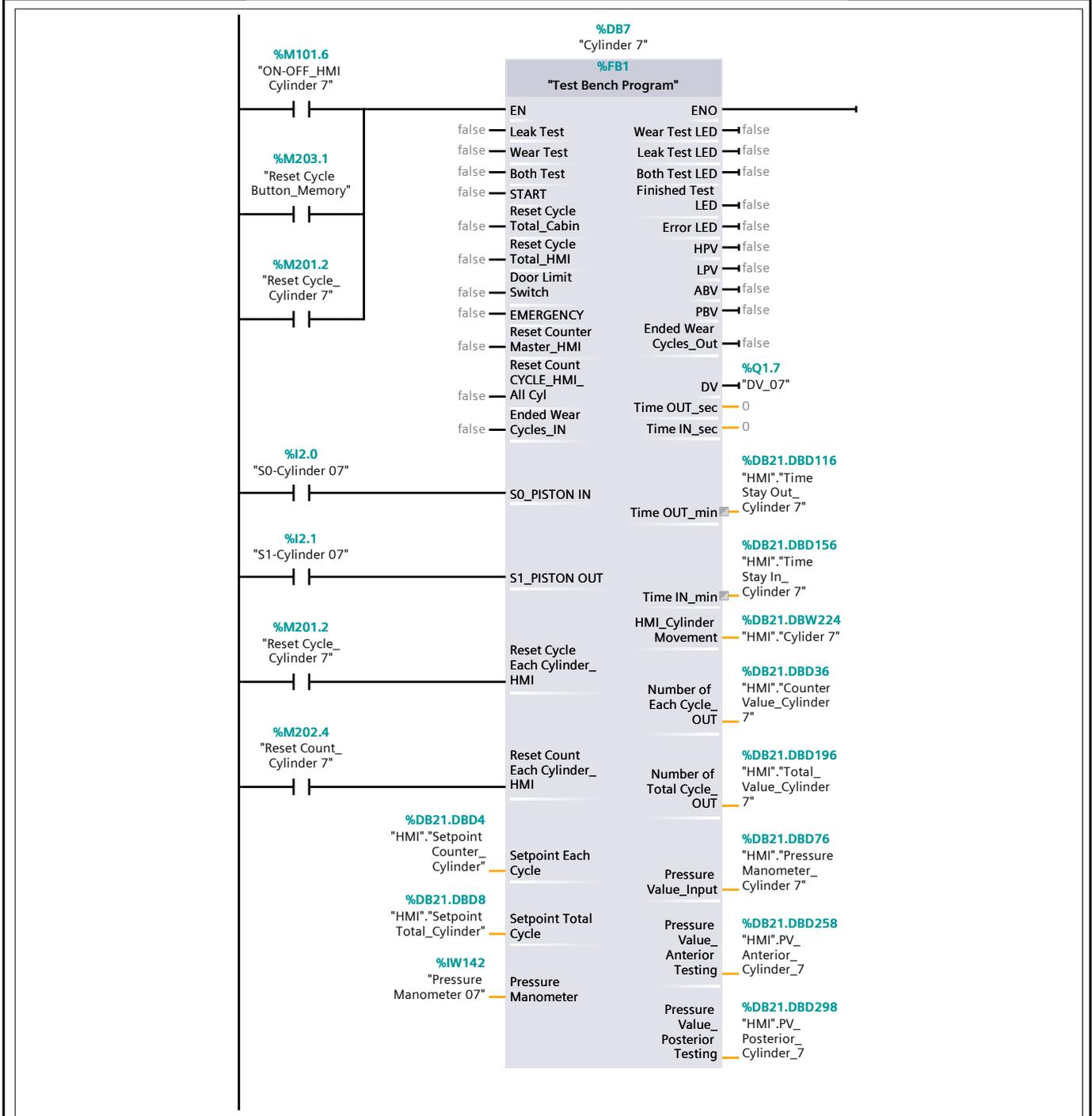
Network 5: Cylinder 5



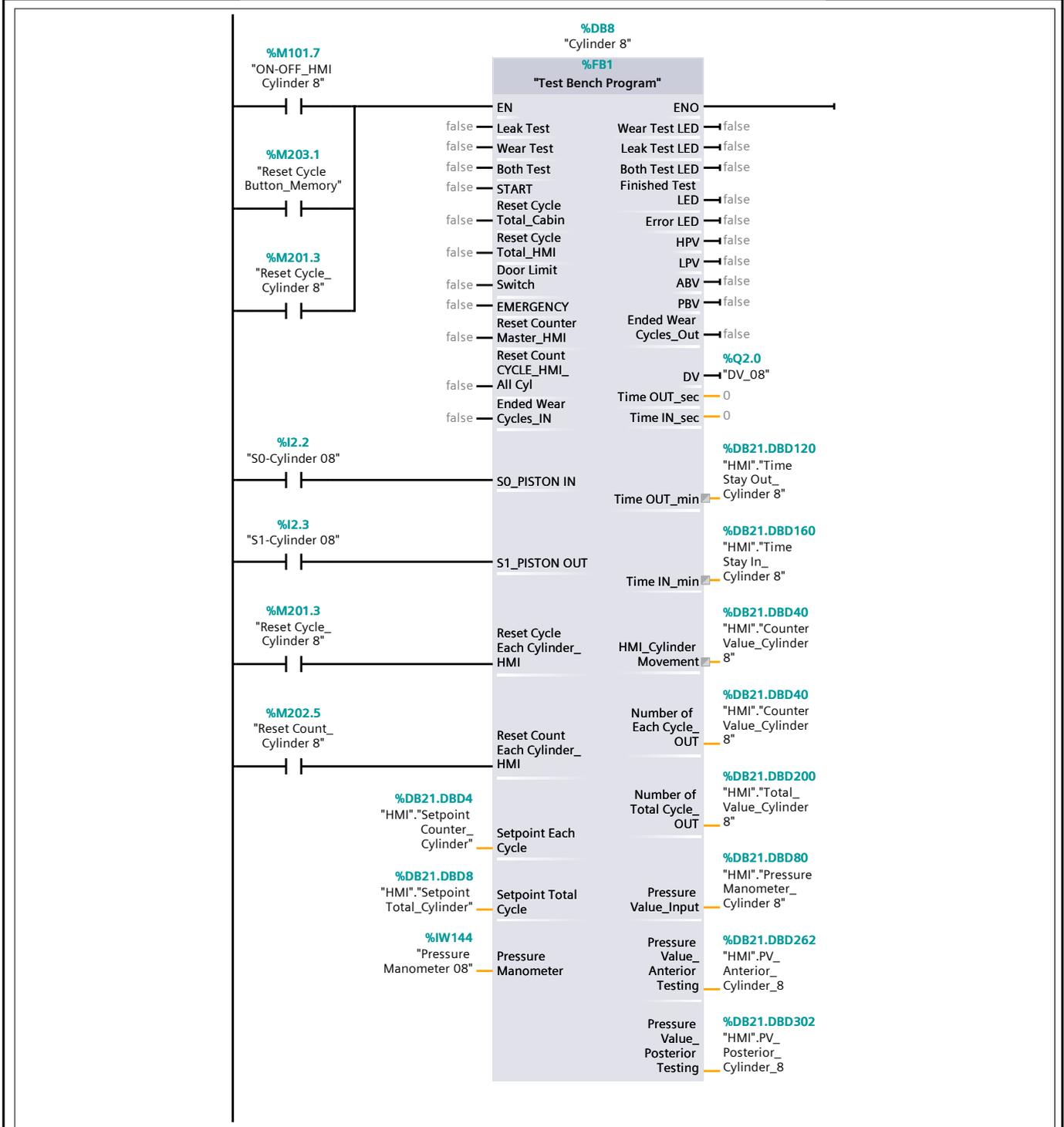
Network 6: Cylinder 6



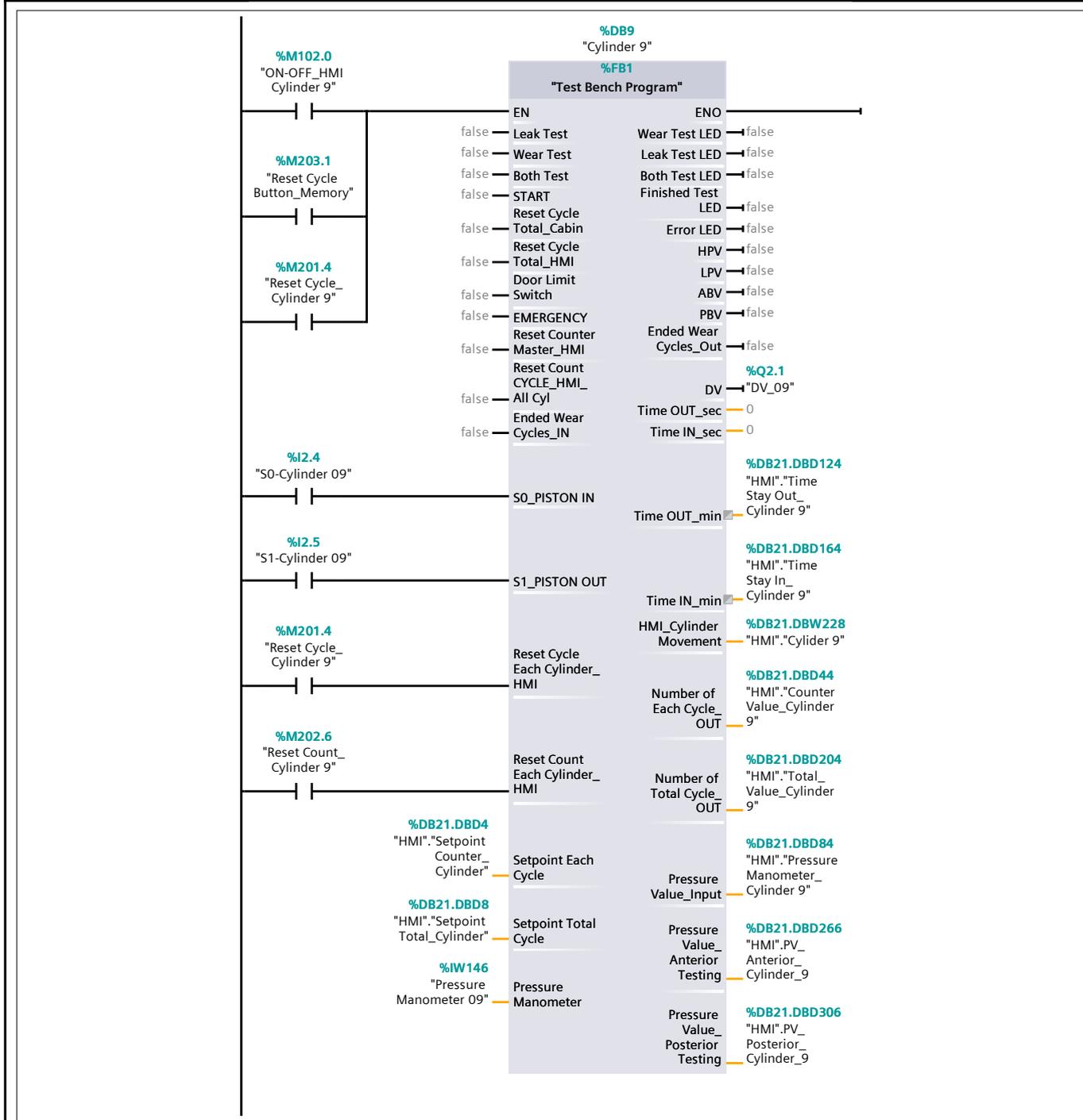
Network 7: Cylinder 7



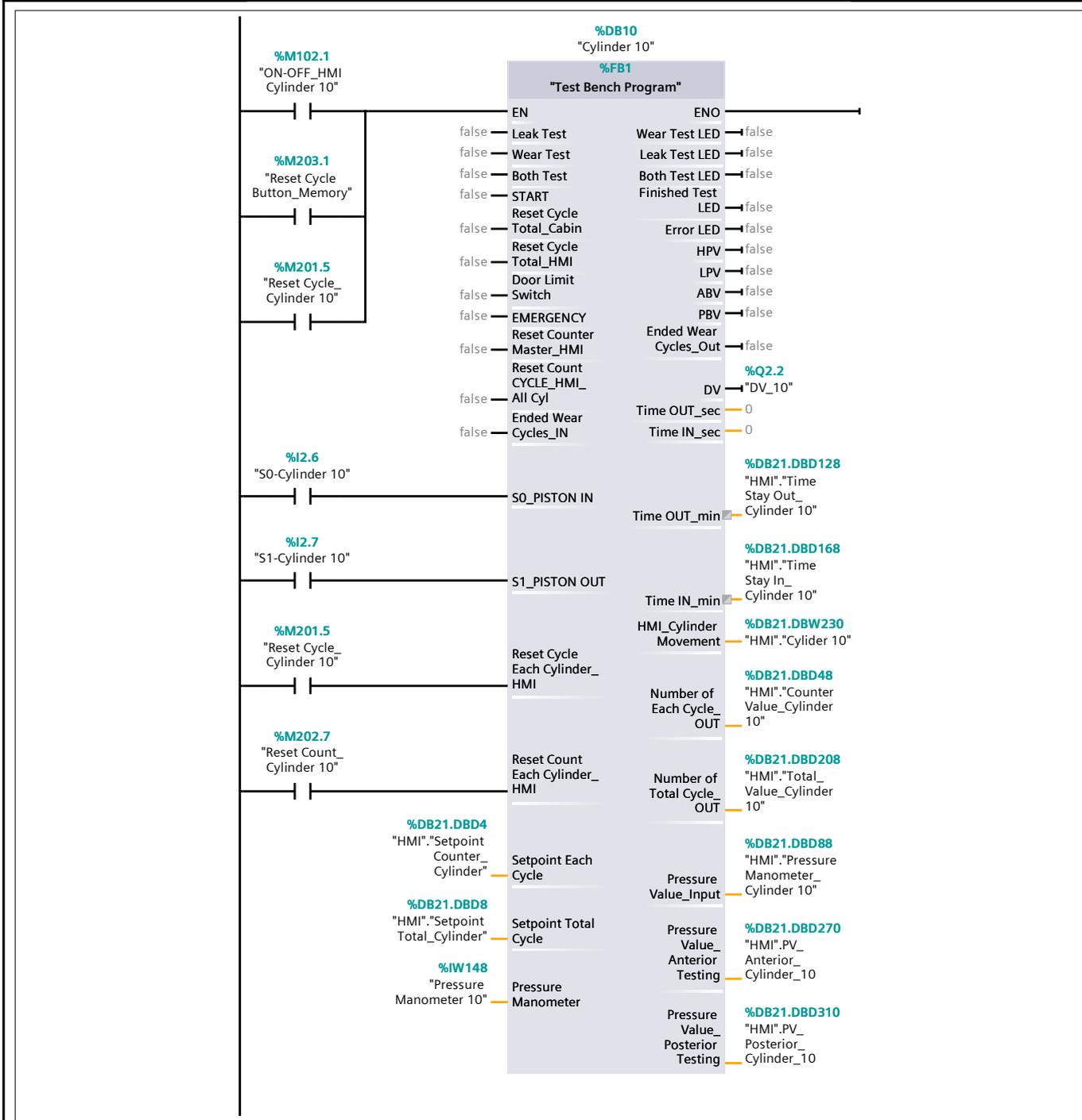
Network 8: Cylinder 8



Network 9: Cylinder 9

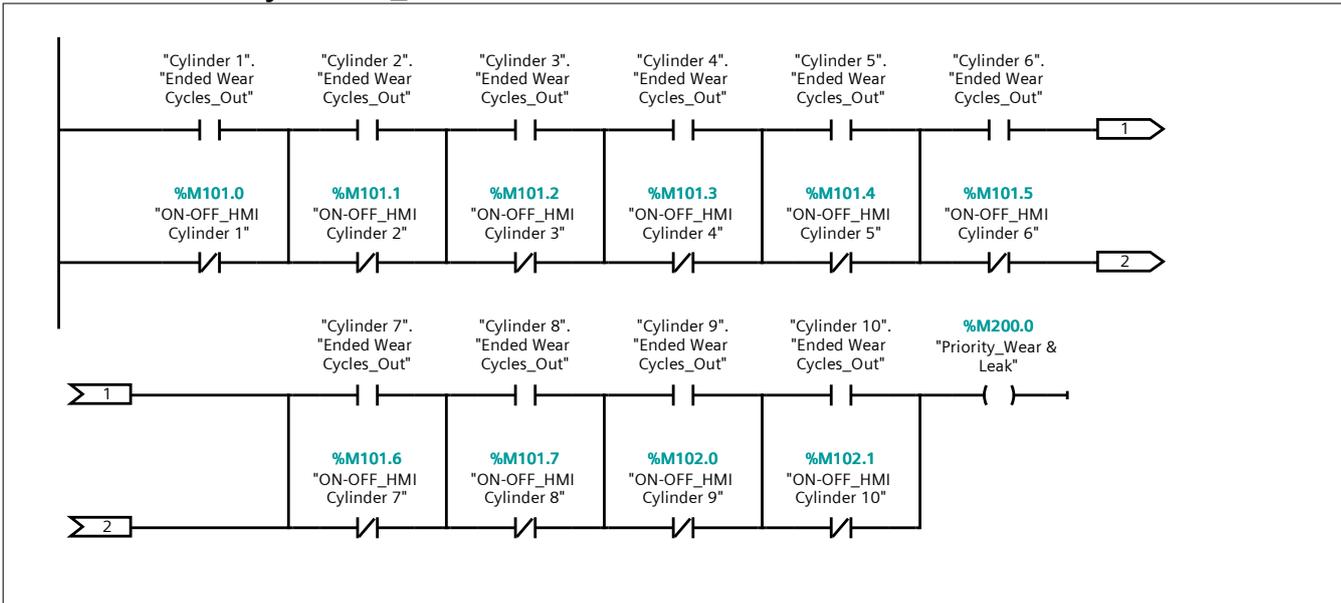


Network 10: Cylinder 10



Network 11: Priority Out Set_of Wear & Leak come from all blocks

Network 11: Priority Out_Set_of Wear & Leak come from all blocks



Network 12: Priority In_to Leak & Wear go to all blocks

%M200.0
"Priority_Wear & Leak"

"Cylinder 1".
"Ended Wear
Cycles_IN"

"Cylinder 2".
"Ended Wear
Cycles_IN"

"Cylinder 3".
"Ended Wear
Cycles_IN"

"Cylinder 4".
"Ended Wear
Cycles_IN"

"Cylinder 5".
"Ended Wear
Cycles_IN"

"Cylinder 6".
"Ended Wear
Cycles_IN"

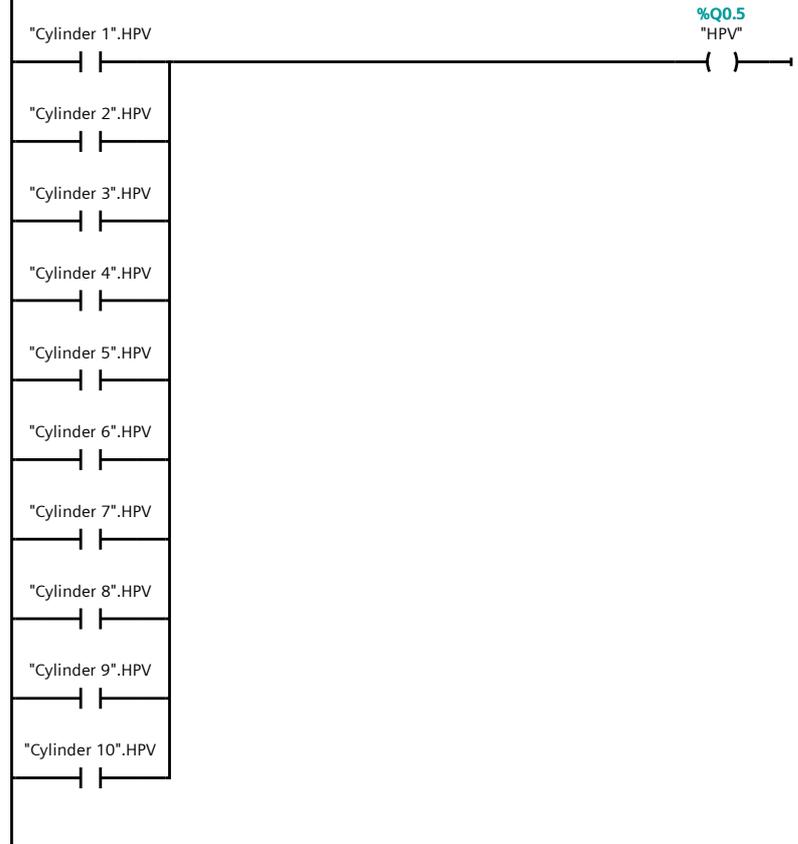
"Cylinder 7".
"Ended Wear
Cycles_IN"

"Cylinder 8".
"Ended Wear
Cycles_IN"

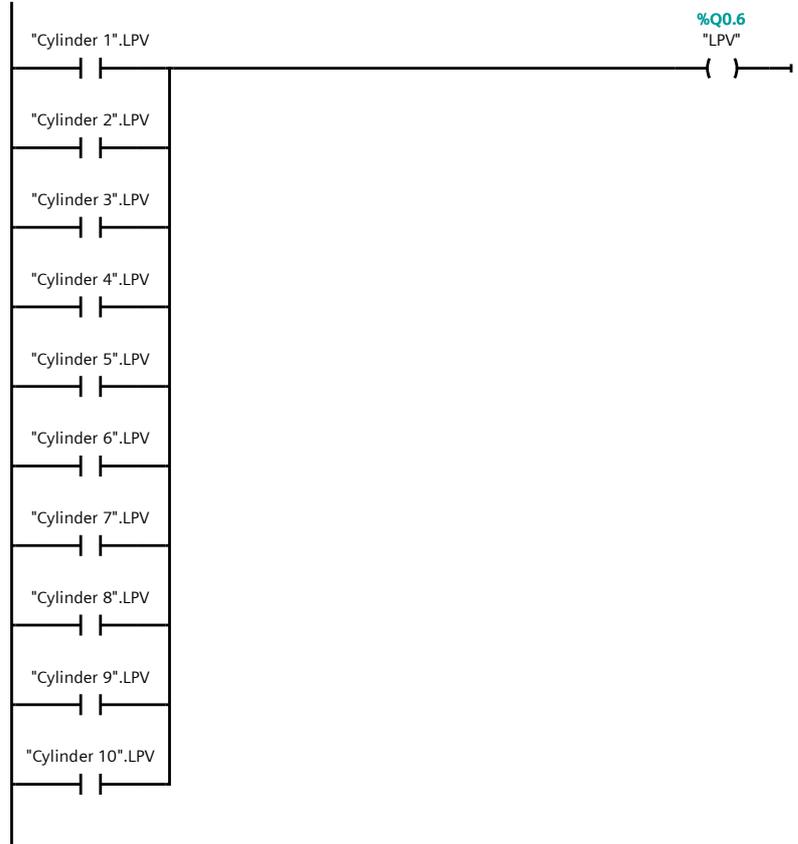
"Cylinder 9".
"Ended Wear
Cycles_IN"

"Cylinder 10".
"Ended Wear
Cycles_IN"

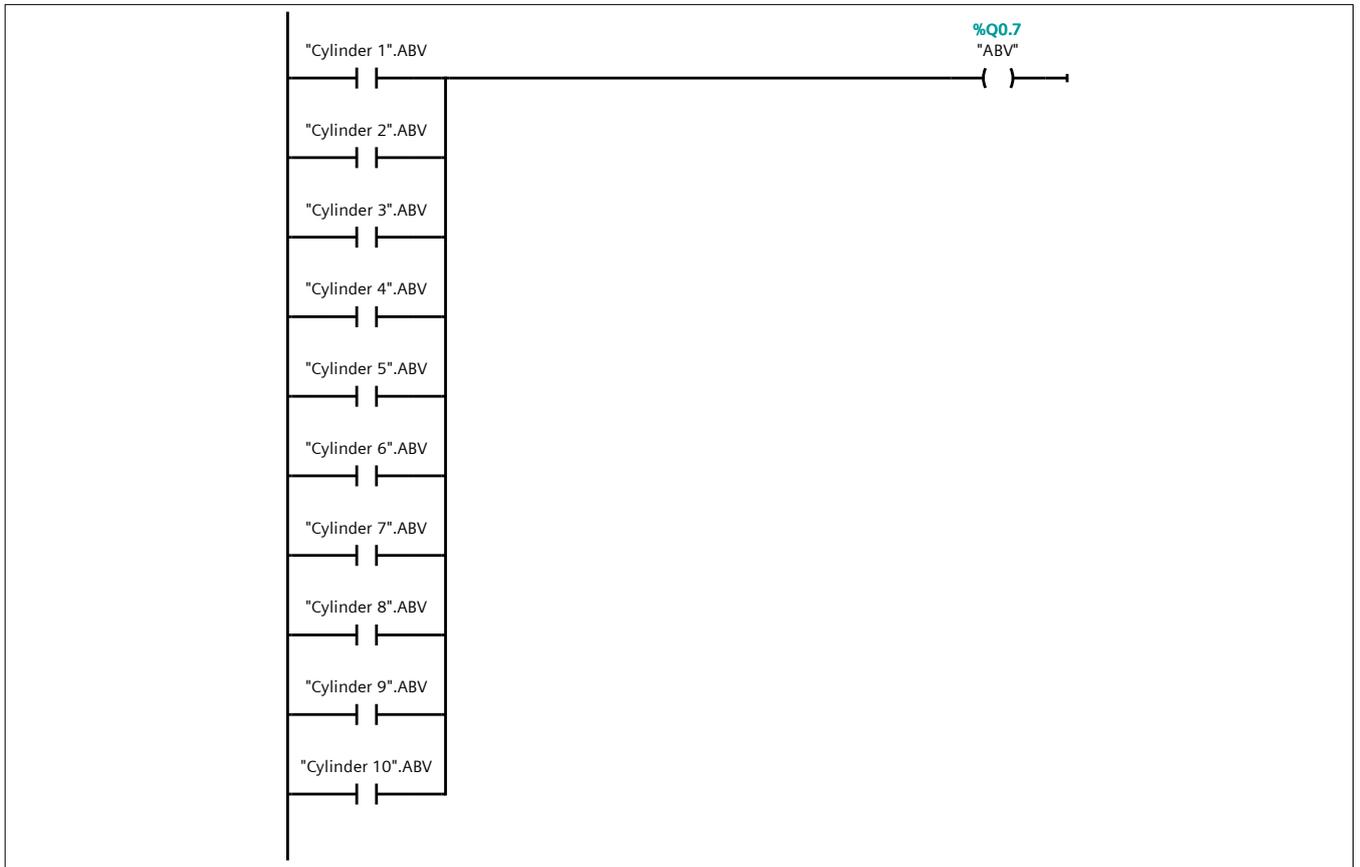
Network 13: High Pressure Valve (HPV)_Output of all cylinders



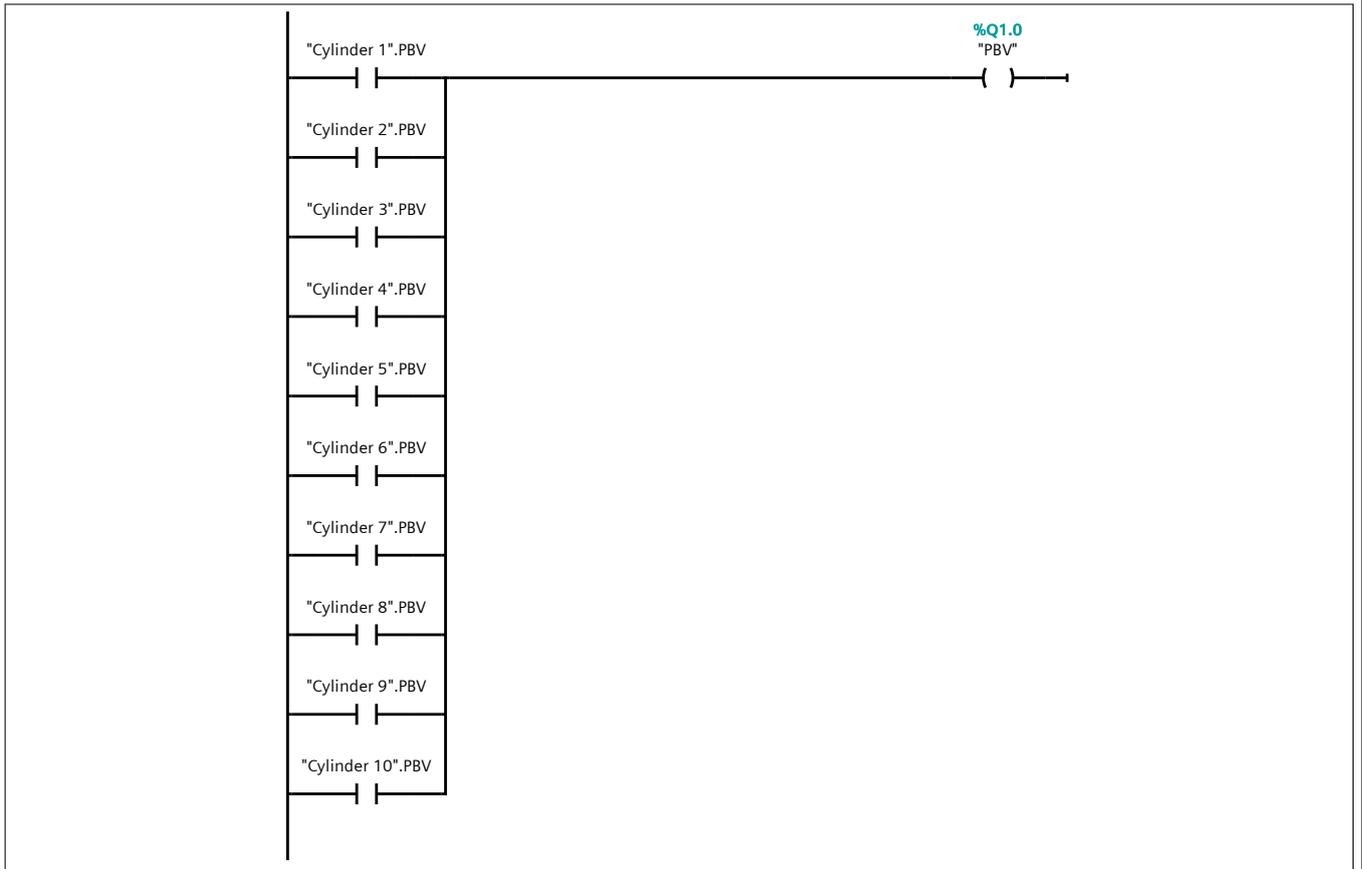
Network 14: Low Pressure Valve (LPV)_Output of all cylinders



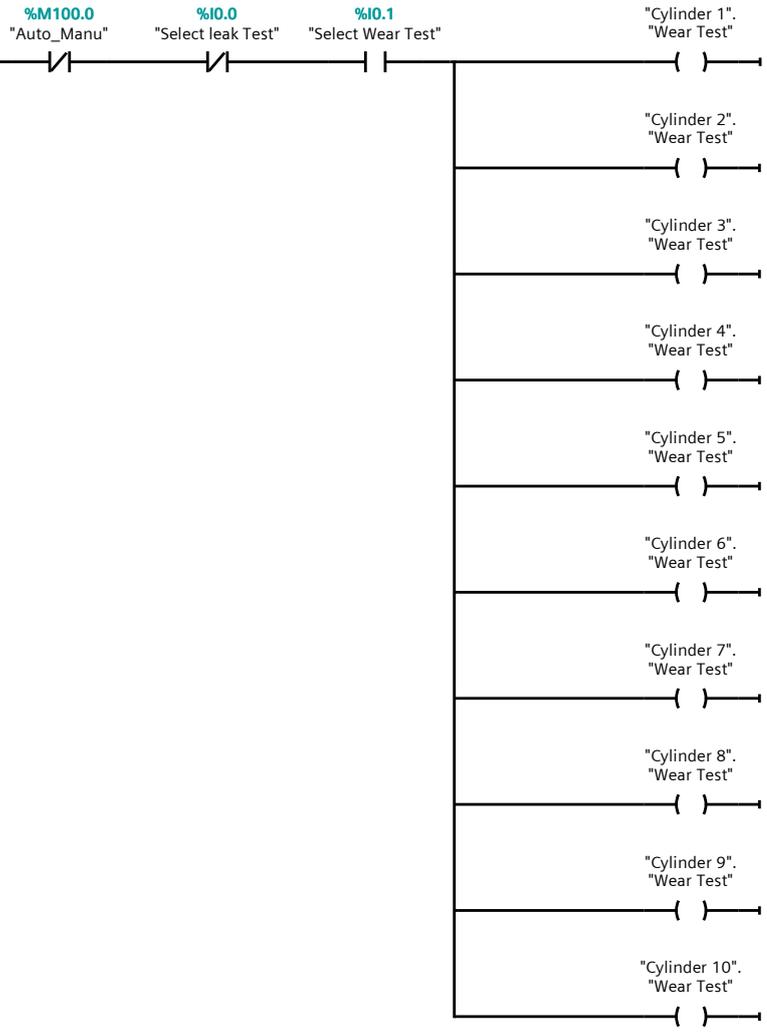
Network 15: Anterior Block Valve (ABV)_Output of all cylinders



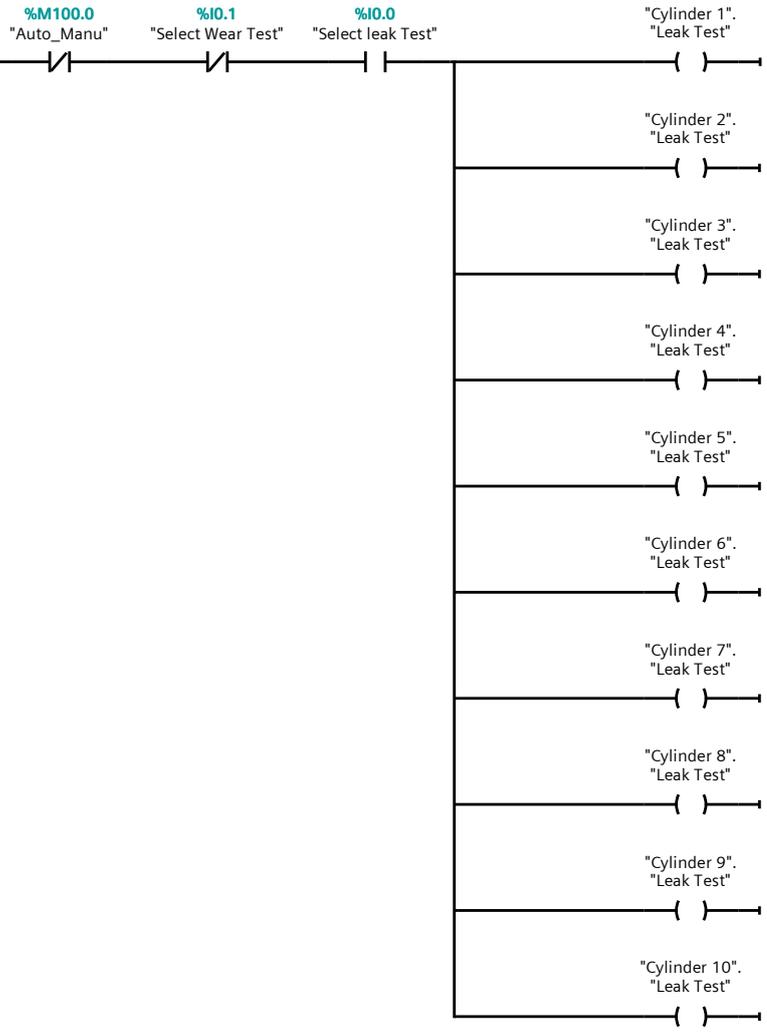
Network 16: Posterior Block Valve (PBV)_Output of all cylinders



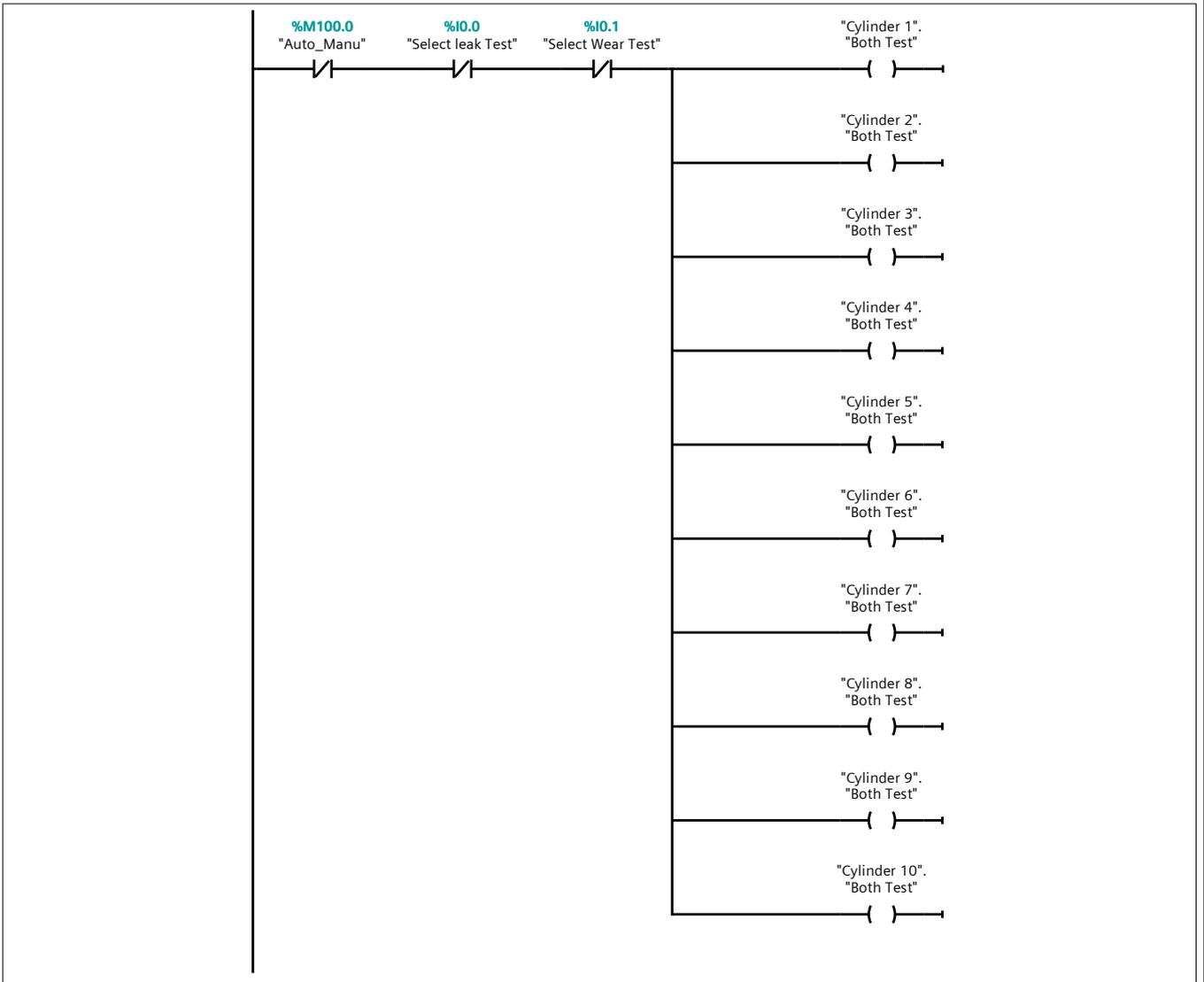
Network 17: Input to "Wear select Mode" of all blocks



Network 18: Input to "Leak select Mode" of all blocks



Network 19: Input to "Both select Mode" of all blocks



Network 20: "Door Limit Switch Button" to all "Door Limit Switch" Blocks' cylinders

%I3.0
"Door Limit
Switch"

"Cylinder 1".
"Door Limit
Switch"

"Cylinder 2".
"Door Limit
Switch"

"Cylinder 3".
"Door Limit
Switch"

"Cylinder 4".
"Door Limit
Switch"

"Cylinder 5".
"Door Limit
Switch"

"Cylinder 6".
"Door Limit
Switch"

"Cylinder 7".
"Door Limit
Switch"

"Cylinder 8".
"Door Limit
Switch"

"Cylinder 9".
"Door Limit
Switch"

"Cylinder 10".
"Door Limit
Switch"

Network 21: "Emergency Button" to all "Emergency" Blocks' cylinders

%I3.1
"Emergency"

"Cylinder 1".
EMERGENCY

"Cylinder 2".
EMERGENCY

"Cylinder 3".
EMERGENCY

"Cylinder 4".
EMERGENCY

"Cylinder 5".
EMERGENCY

"Cylinder 6".
EMERGENCY

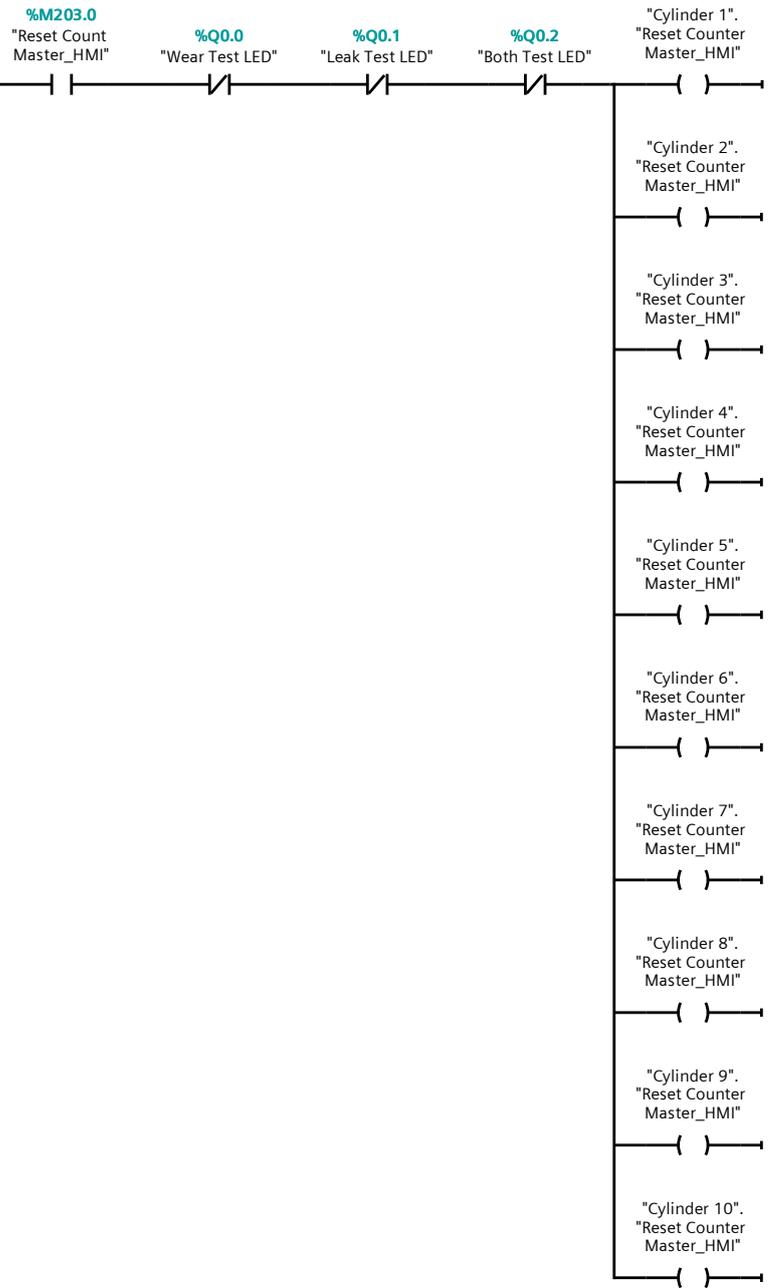
"Cylinder 7".
EMERGENCY

"Cylinder 8".
EMERGENCY

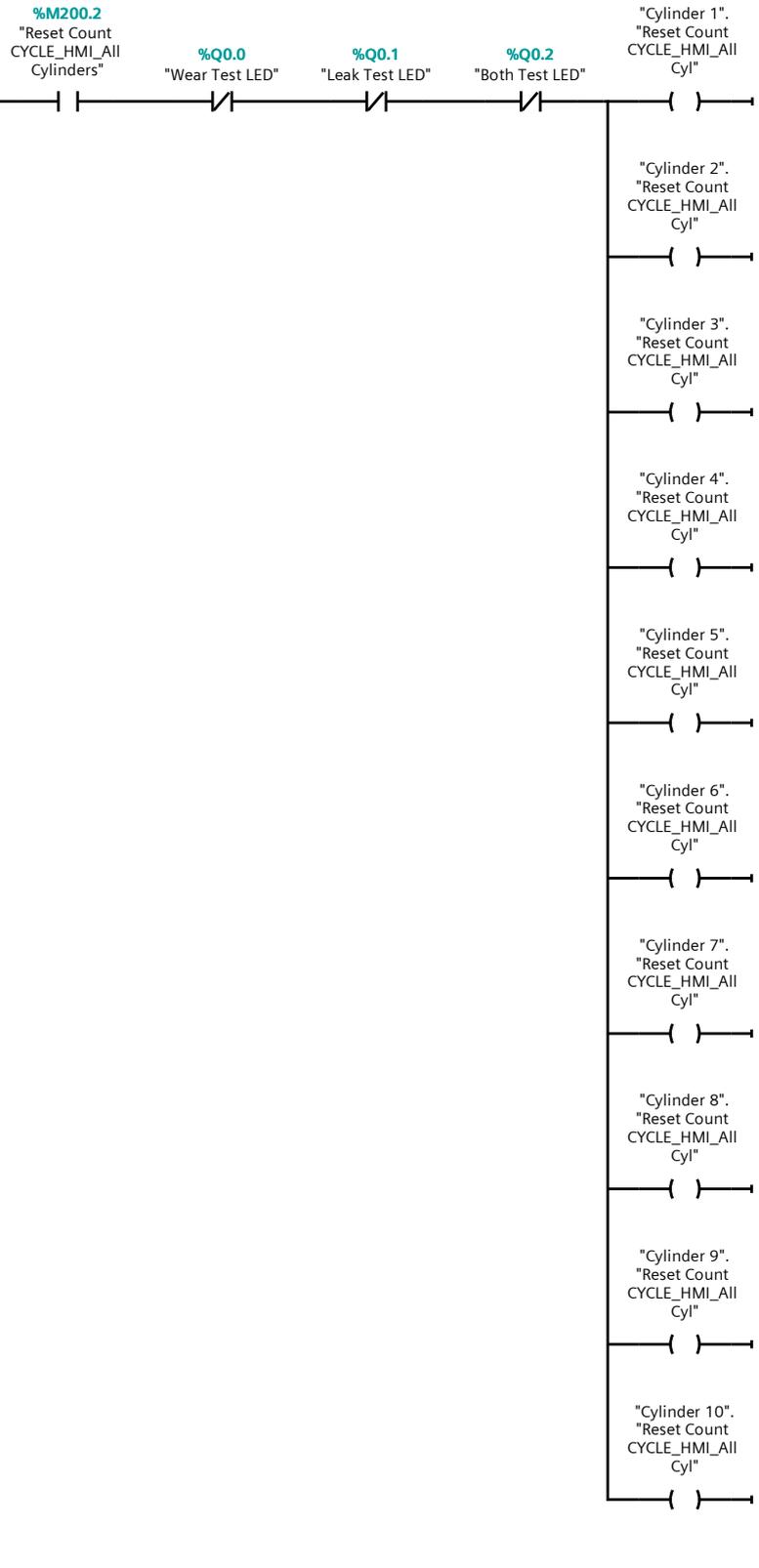
"Cylinder 9".
EMERGENCY

"Cylinder 10".
EMERGENCY

Network 22: "Reset Count Master_HMI Button" to all "Reset Count master_HMI" Blocks' cylinders



Network 23: Reset Counter Each CYCLE_HMI_one for ALL Cylinders



Network 24: "Reset HMI Button" to all "Reset Cycle" Blocks' cylinders

%M200.3
"Reset Cycle
Total_HMI"

"Cylinder 1".
"Reset Cycle
Total_HMI"

"Cylinder 2".
"Reset Cycle
Total_HMI"

"Cylinder 3".
"Reset Cycle
Total_HMI"

"Cylinder 4".
"Reset Cycle
Total_HMI"

"Cylinder 5".
"Reset Cycle
Total_HMI"

"Cylinder 6".
"Reset Cycle
Total_HMI"

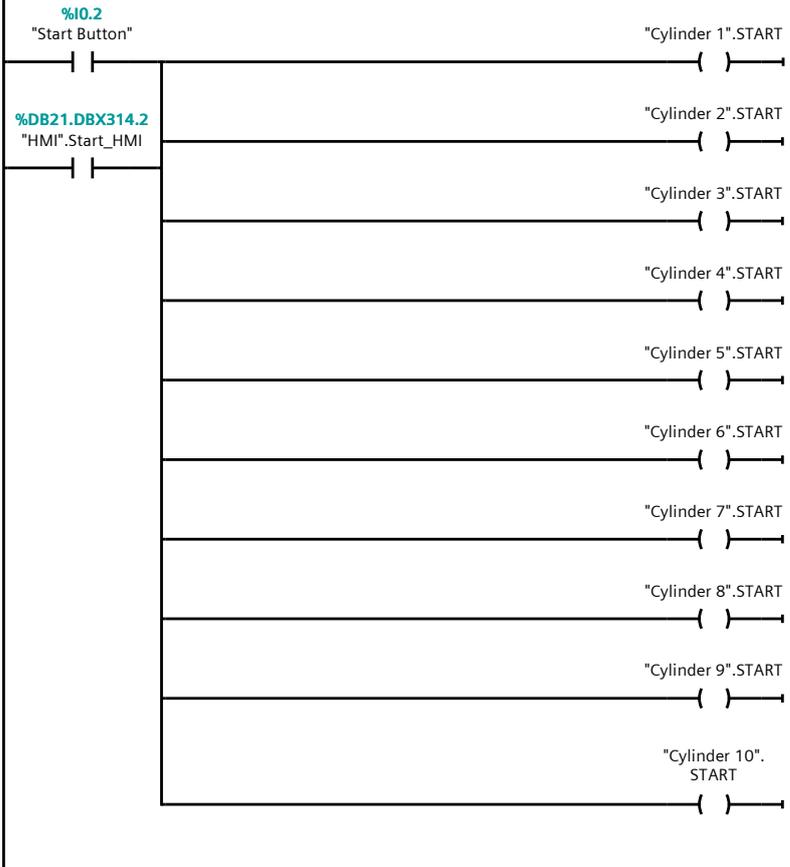
"Cylinder 7".
"Reset Cycle
Total_HMI"

"Cylinder 8".
"Reset Cycle
Total_HMI"

"Cylinder 9".
"Reset Cycle
Total_HMI"

"Cylinder 10".
"Reset Cycle
Total_HMI"

Network 25: "Start Button Cabin" & "Start Button HMI" to all "Start" Blocks' cylinders



Network 26: "Reset cabin Button" to all "Reset Cycle" Blocks' cylinders

%M203.1
"Reset Cycle
Button_Memory"

"Cylinder 1".
"Reset Cycle
Total_Cabin"

"Cylinder 2".
"Reset Cycle
Total_Cabin"

"Cylinder 3".
"Reset Cycle
Total_Cabin"

"Cylinder 4".
"Reset Cycle
Total_Cabin"

"Cylinder 5".
"Reset Cycle
Total_Cabin"

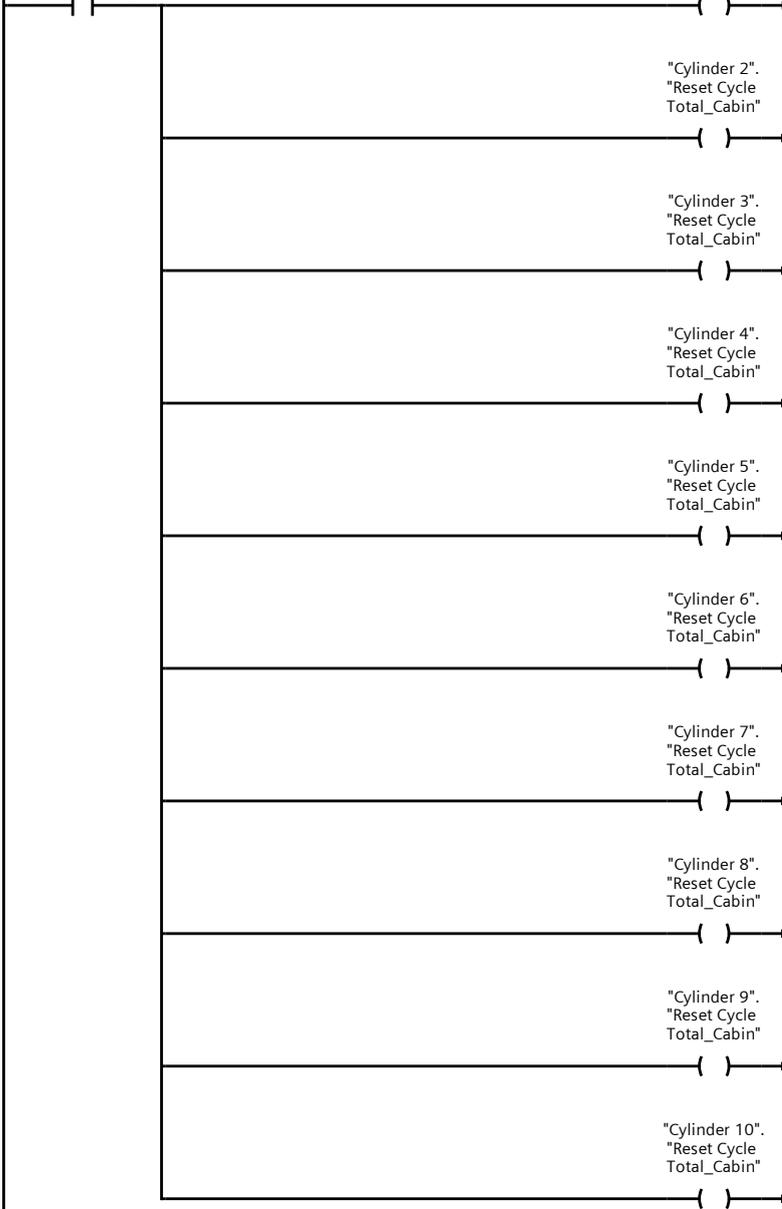
"Cylinder 6".
"Reset Cycle
Total_Cabin"

"Cylinder 7".
"Reset Cycle
Total_Cabin"

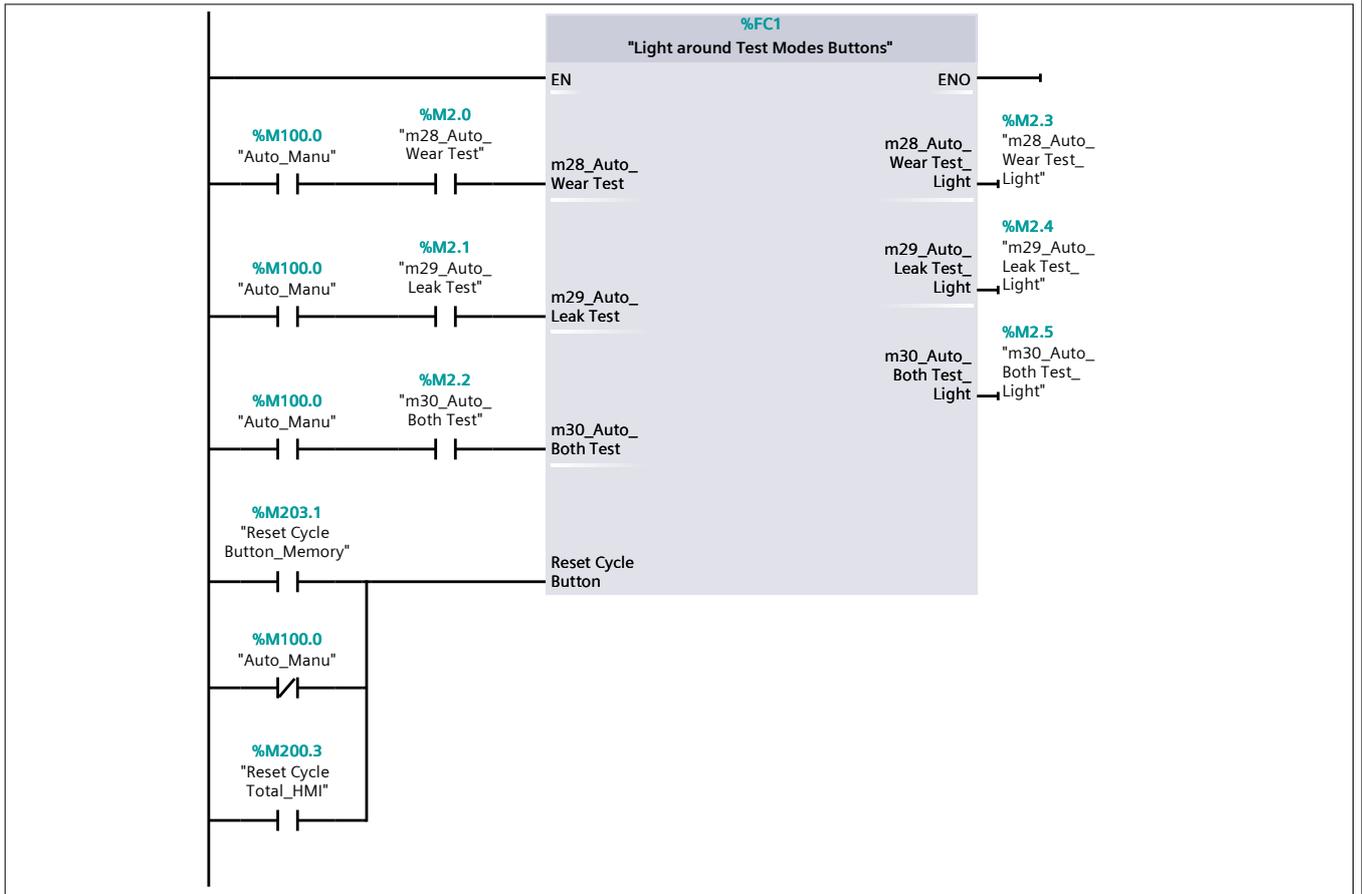
"Cylinder 8".
"Reset Cycle
Total_Cabin"

"Cylinder 9".
"Reset Cycle
Total_Cabin"

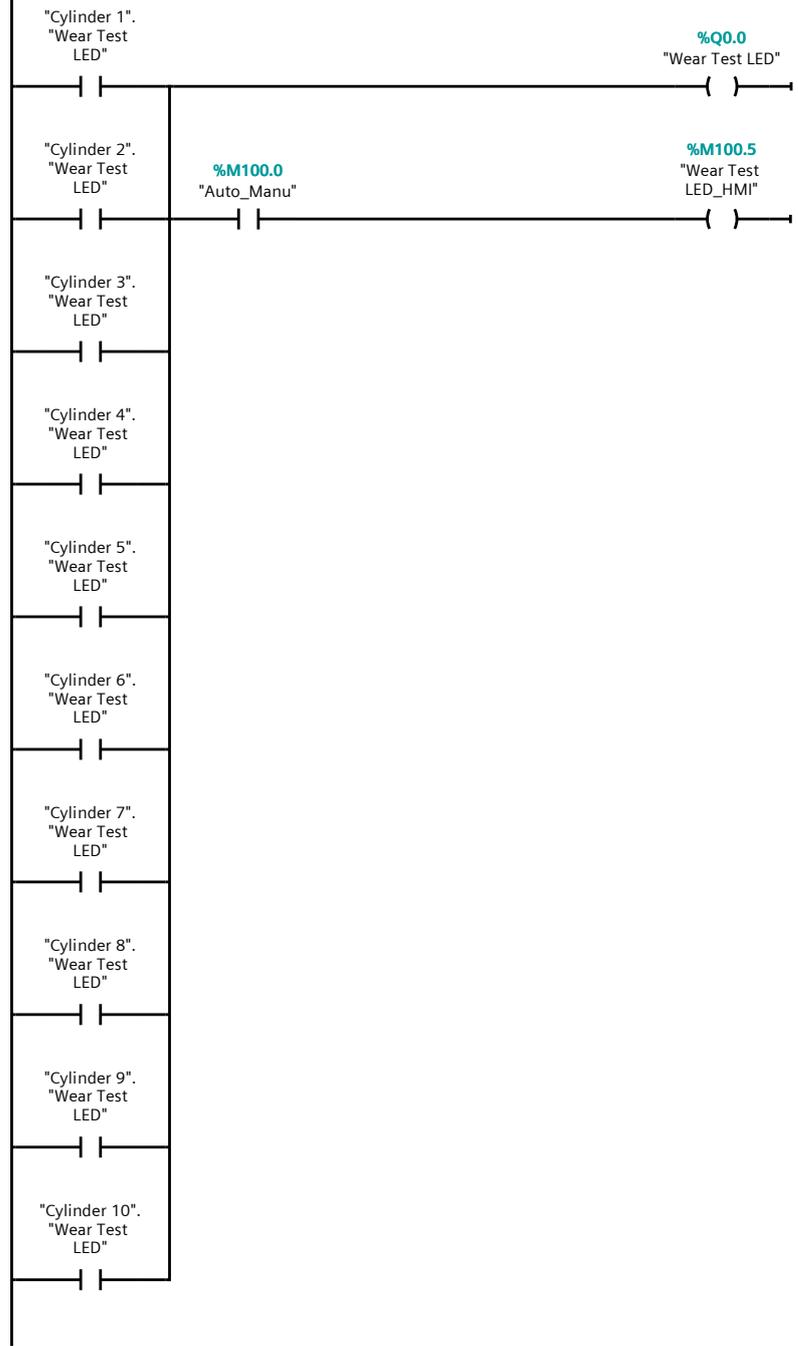
"Cylinder 10".
"Reset Cycle
Total_Cabin"



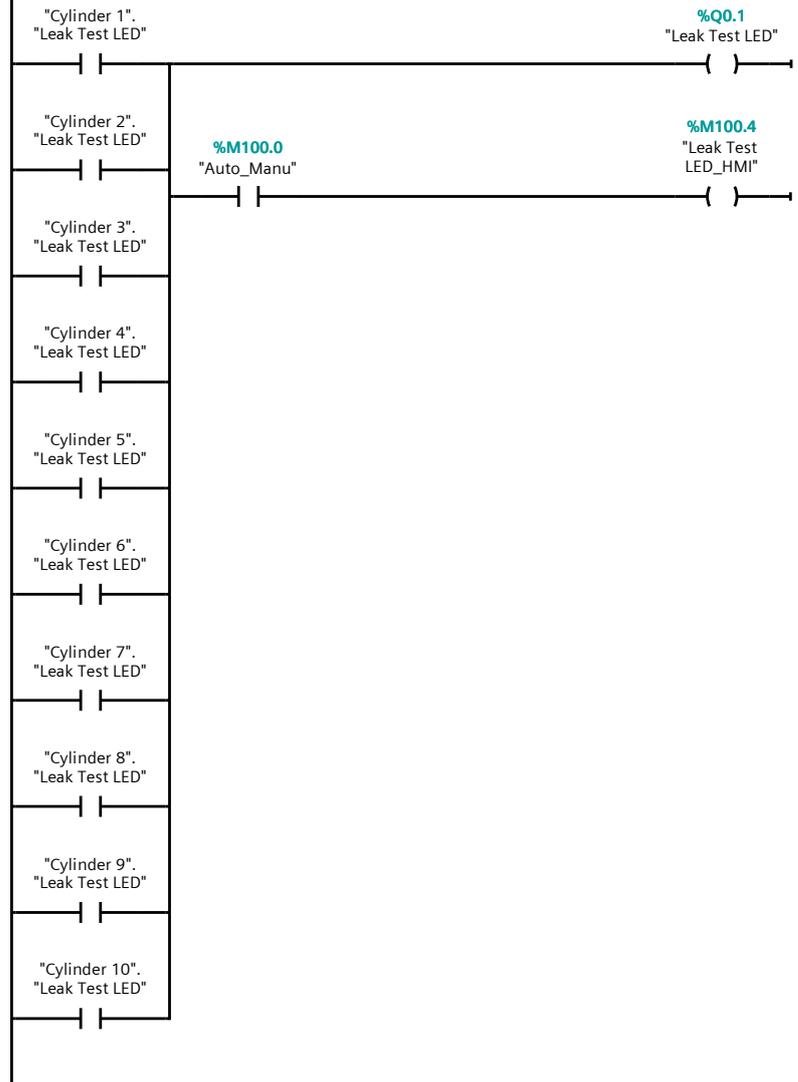
Network 27: Selector Test Mode Buttons on HMI



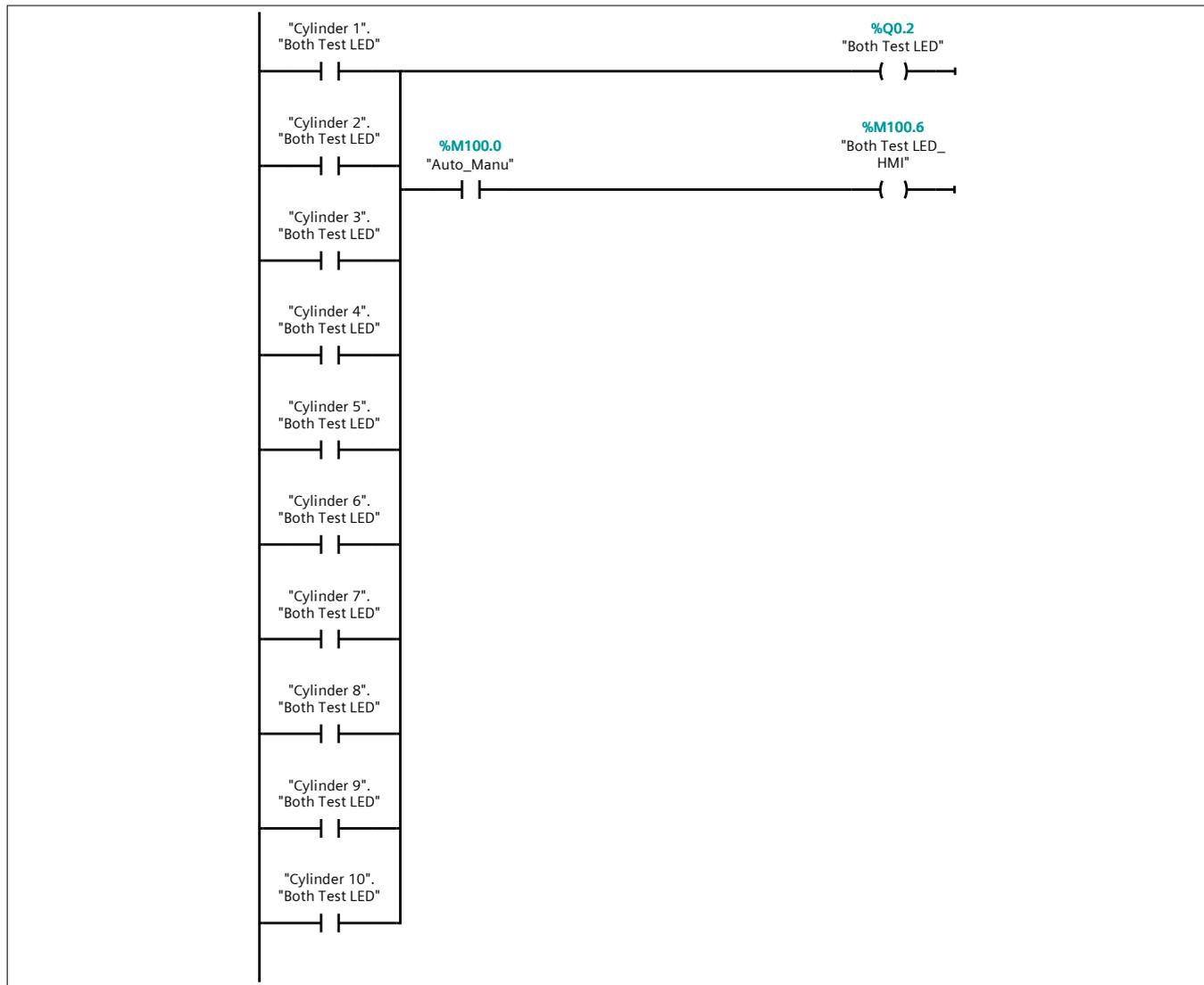
Network 29: "Wear Test LED" _Output of all cylinders



Network 30: "Leak Test LED"_Output of all cylinders

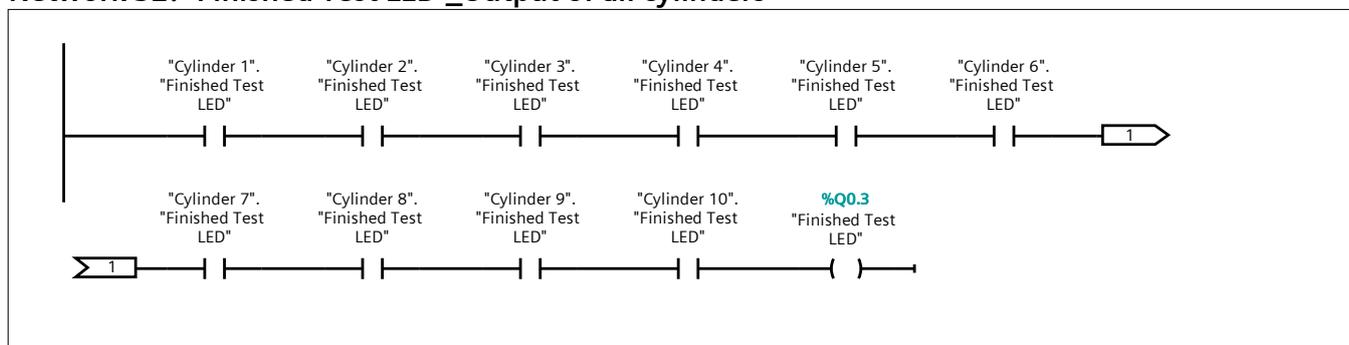


Network 31: "Both Test LED"_Output of all cylinders

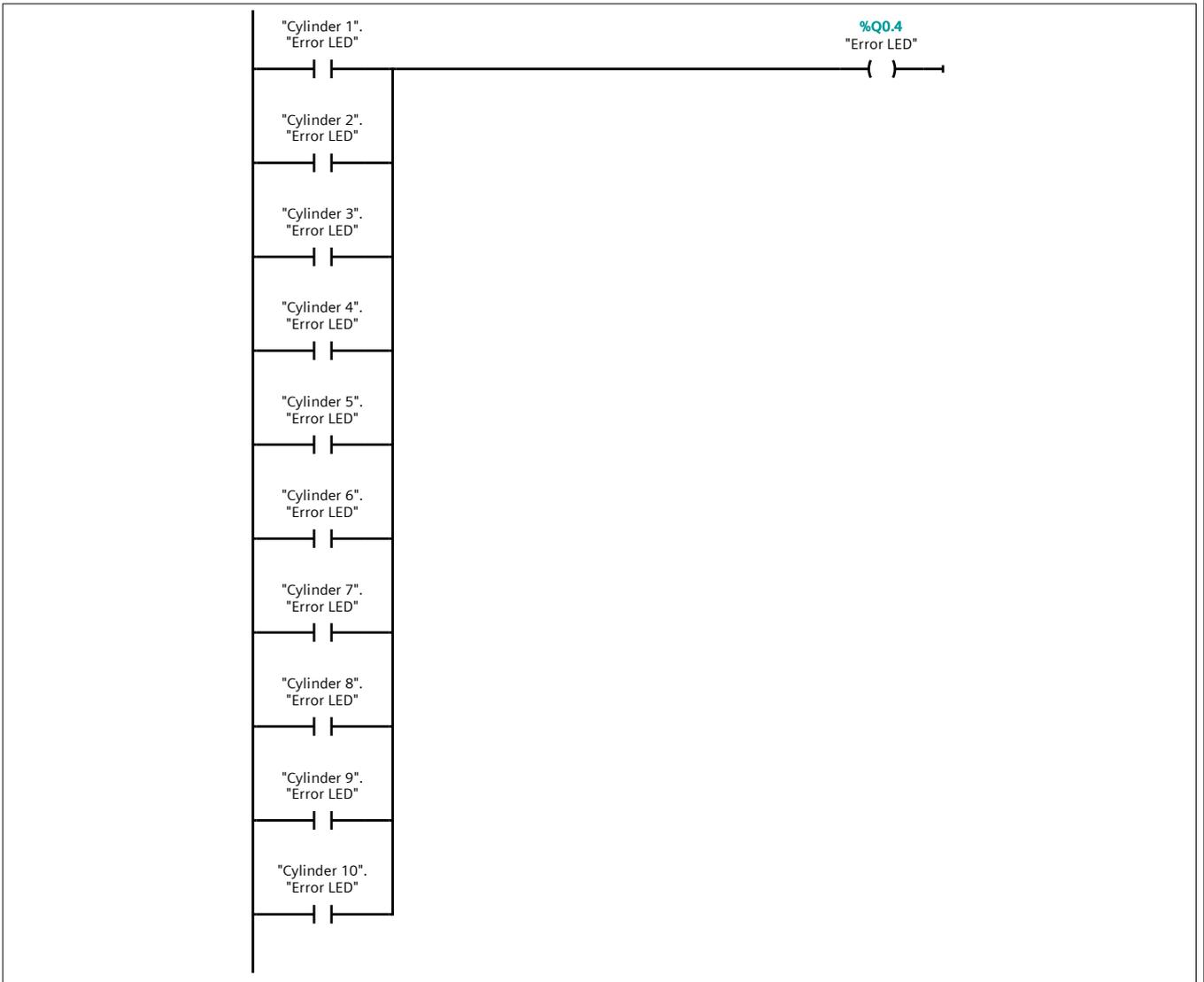


Network 32: "Finished Test LED"_Output of all cylinders

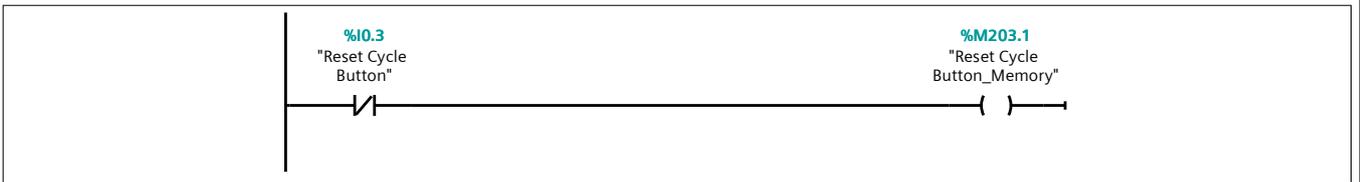
Network 32: "Finished Test LED"_Output of all cylinders



Network 33: "Error LED"__Output of all cylinders



Network 34: Memory for Input Reset Button_Cabin



Here is the end of Main OB1 Block

Automation for All Cyl_22 / PLC_6 [CPU 1512C-1 PN] / Program blocks

Test Bench Program [FB1]

Test Bench Program Properties

General

Name	Test Bench Program	Number	1	Type	FB
Language	LAD	Numbering	Automatic		

Information

Title	Automation programming for Lifetime measurement of Pneumatic cylinder_Designed & Implemented by MOHAMMAD SHEKASTEH	Author		Comment	Automation and Programming for measuring the Pneumatic Cylinders' Lifetime by PLC S7-1500_ in TIA Portal software
Family		Version	0.1	User-defined ID	

Test Bench Program

Name	Data type	Default value	Retain	Accessible from HMI/OP C UA	Writable from HMI/OP C UA	Visible in engineering	Set-point	Supervision	Comment
▼ Input									
Leak Test	Bool	false	Non-retain	True	True	True	False		
Wear Test	Bool	false	Non-retain	True	True	True	False		
Both Test	Bool	false	Non-retain	True	True	True	False		
START	Bool	false	Non-retain	True	True	True	False		
Reset Cycle Total_Cabin	Bool	false	Non-retain	True	True	True	False		
Reset Cycle Total_HMI	Bool	false	Non-retain	True	True	True	False		
Door Limit Switch	Bool	false	Non-retain	True	True	True	False		
EMERGENCY	Bool	false	Non-retain	True	True	True	False		
Reset Counter Master_HMI	Bool	false	Non-retain	True	True	True	False		
Reset Count CYCLE_HMI_All Cyl	Bool	false	Non-retain	True	True	True	False		
Ended Wear Cycles_IN	Bool	false	Non-retain	True	True	True	False		
S0_PISTON IN	Bool	false	Non-retain	True	True	True	False		
S1_PISTON OUT	Bool	false	Non-retain	True	True	True	False		
Reset Cycle Each Cylinder_HMI	Bool	false	Non-retain	True	True	True	False		
Reset Count Each Cylinder_HMI	Bool	false	Non-retain	True	True	True	False		

Totally Integrated Automation Portal									
Name	Data type	Default value	Retain	Access-ible from HMI/OP C UA	Wri-ta-ble from HM I/O PC UA	Visible in HMI engi-neer-ing	Set-point	Super- vision	Comment
Setpoint Each Cycle	DInt	0	Non-retain	True	True	True	False		
Setpoint Total Cycle	DInt	0	Non-retain	True	True	True	False		
Pressure Manometer	Int	0	Non-retain	True	True	True	False		
▼ Output									
Wear Test LED	Bool	false	Non-retain	True	True	True	False		
Leak Test LED	Bool	false	Non-retain	True	True	True	False		
Both Test LED	Bool	false	Non-retain	True	True	True	False		
Finished Test LED	Bool	false	Non-retain	True	True	True	False		
Error LED	Bool	false	Non-retain	True	True	True	False		
HPV	Bool	false	Non-retain	True	True	True	False		
LPV	Bool	false	Non-retain	True	True	True	False		
ABV	Bool	false	Non-retain	True	True	True	False		
PBV	Bool	false	Non-retain	True	True	True	False		
Ended Wear Cycles_Out	Bool	false	Non-retain	True	True	True	False		
DV	Bool	false	Non-retain	True	True	True	False		
Time OUT_sec	DInt	0	Non-retain	True	True	True	False		
Time IN_sec	DInt	0	Non-retain	True	True	True	False		
Time OUT_min	DInt	0	Non-retain	True	True	True	False		
Time IN_min	DInt	0	Non-retain	True	True	True	False		
HMI_Cylinder Movement	Int	0	Non-retain	True	True	True	False		
Number of Each Cycle_OUT	DInt	0	Non-retain	True	True	True	False		
Number of Total Cycle_OUT	DInt	0	Non-retain	True	True	True	False		
Pressure Value_Input	Real	0.0	Non-retain	True	True	True	False		
Pressure Value_Anterior Testing	Real	0.0	Non-retain	True	True	True	False		
Pressure Value_Posterior Testing	Real	0.0	Non-retain	True	True	True	False		
InOut									

Totally Integrated Automation Portal									
Name	Data type	Default value	Retain	Access-ible from HMI/OP C UA	Wri-ta-ble from HM I/O PC UA	Visible in HMI engi-neer-ing	Set-point	Super- vision	Comment
▼ Static									
m1_Leak Test	Bool	false	Non-retain	True	True	True	False		Set Leak Test
m2	Bool	false	Non-retain	True	True	True	False		In Leak_Move OUT
m3	Bool	false	Non-retain	True	True	True	False		
m4	Bool	false	Non-retain	True	True	True	False		Measuring_Posterior Chamber Leak Test_for 30 Min
m5	Bool	false	Non-retain	True	True	True	False		
m6	Bool	false	Non-retain	True	True	True	False		
m7	Bool	false	Non-retain	True	True	True	False		Measuring_Front Chamber Leak Test_for 30 Min
m8	Bool	false	Non-retain	True	True	True	False		
m15	Bool	false	Non-retain	True	True	True	False		
m16	Bool	false	Non-retain	True	True	True	False		
m10_Wear Test	Bool	false	Non-retain	True	True	True	False		Set Wear Test
m11	Bool	false	Non-retain	True	True	True	False		In Wear_Move OUT
m12	Bool	false	Non-retain	True	True	True	False		In Wear_Move IN
m13	Bool	false	Non-retain	True	True	True	False		
m14	Bool	false	Non-retain	True	True	True	False		
m17	Bool	false	Non-retain	True	True	True	False		
m18-Both Test	Bool	false	Non-retain	True	True	True	False		
m19	Bool	false	Non-retain	True	True	True	False		
m20	Bool	false	Non-retain	True	True	True	False		
m21	Bool	false	Non-retain	True	True	True	False		
m22	Bool	false	Non-retain	True	True	True	False		
m23	Bool	false	Non-retain	True	True	True	False		Automatic Restart Wear in Both Mode
m24	Bool	false	Non-retain	True	True	True	False		Automatic Restart Leak in Both Mode
m25	Bool	false	Non-retain	True	True	True	False		
m26	Bool	false	Non-retain	True	True	True	False		

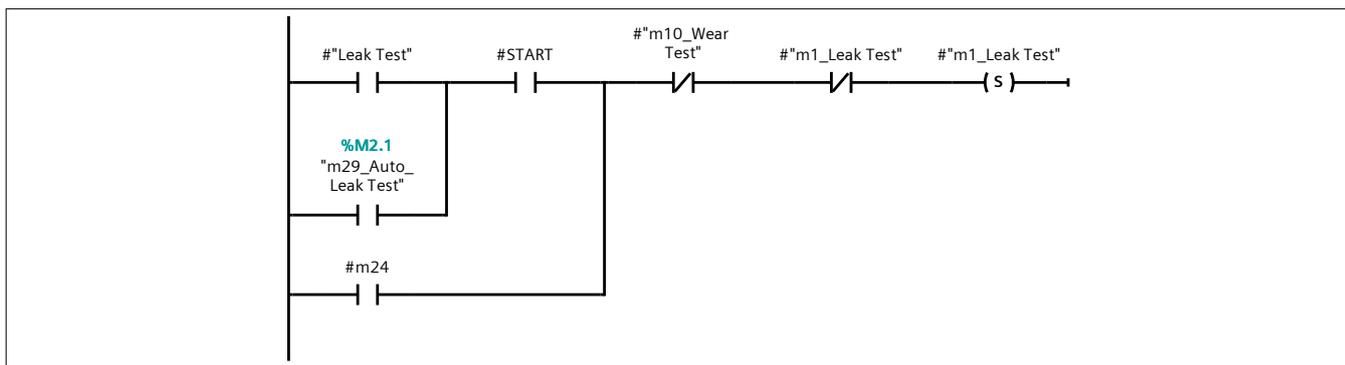
Name	Data type	Default value	Retain	Access-ible from HMI/OP C UA	Wri-table from HM I/O PC UA	Visible in HMI engineering	Set-point	Super- vision	Comment
m27	Bool	false	Non-retain	True	True	True	False		
m28_Limit Course_W	Bool	false	Non-retain	True	True	True	False		Memory set_Wear_for Time Limitation Course
m29_Limit Course_L	Bool	false	Non-retain	True	True	True	False		Memory set_Leak_for Time Limitation Course
m30_Sens_Problem	Bool	false	Non-retain	True	True	True	False		Sensor Problems
m31_pressure Error	Bool	false	Non-retain	True	True	True	False		
Number of Single Cycle	DInt	0	Non-retain	True	True	True	False		
Number of Total Cycles	DInt	0	Non-retain	True	True	True	False		
▼ timer1	IEC_TIMER		Non-retain	True	True	True	True		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer2	IEC_TIMER		Non-retain	True	True	True	True		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer3	IEC_TIMER		Non-retain	True	True	True	True		Piston Remains OUT For 30 Minutes
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer4	IEC_TIMER		Non-retain	True	True	True	True		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		

Totally Integrated Automation Portal									
Name	Data type	Default value	Retain	Accessible from HMI/OPCUA	Writable from HM I/O PCUA	Visible in HMI engineering	Set-point	Supervision	Comment
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer5	IEC_TIMER		Non-retain	True	True	True	True		
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer6	IEC_TIMER		Non-retain	True	True	True	True		Leak_Piston remains Inside for 30 Minutes
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer7	IEC_TIMER		Non-retain	True	True	True	True		Limitation Course in Wear Mode
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ timer8	IEC_TIMER		Non-retain	True	True	True	True		Limitation Course in Leak Mode
PT	Time	T#0ms	Non-retain	True	True	True	False		
ET	Time	T#0ms	Non-retain	True	False	True	False		
IN	Bool	false	Non-retain	True	True	True	False		
Q	Bool	false	Non-retain	True	False	True	False		
▼ Temp									
ms_remain out	DInt								
ms_remain in	DInt								
time Out_sec_1	DInt								
time In_sec_1	DInt								
Always_1	Bool								
RET_VAL	Word								

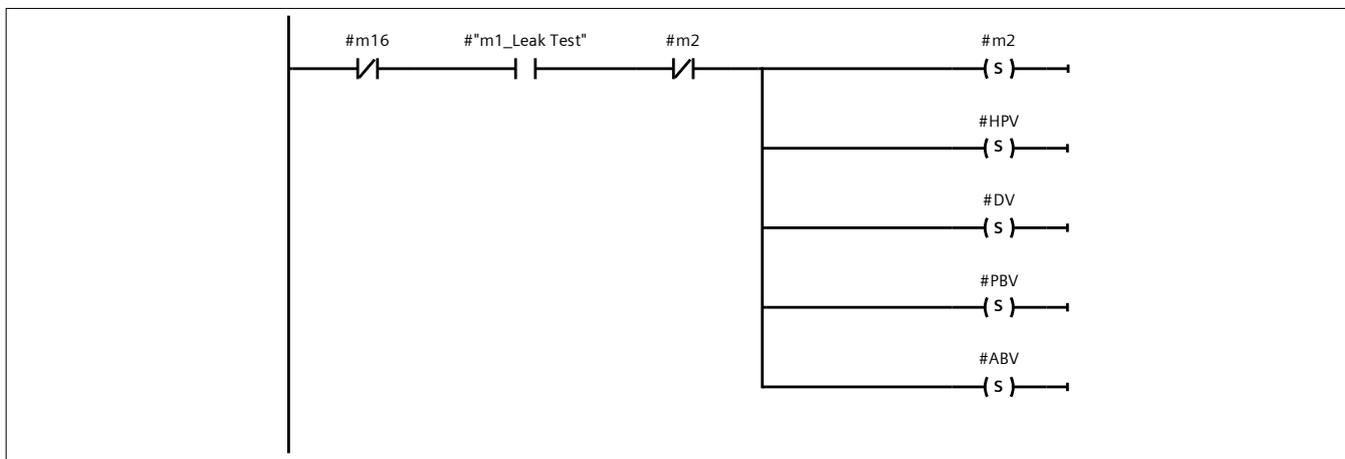
Name	Data type	Default value	Retain	Access-ible from HMI/OP C UA	Wri-ta-ble from HM I/O PC UA	Visible Set- in HMI point engineering	Super- vision	Comment
Constant								

Network 1: Leak_Go to Leak Test Mode

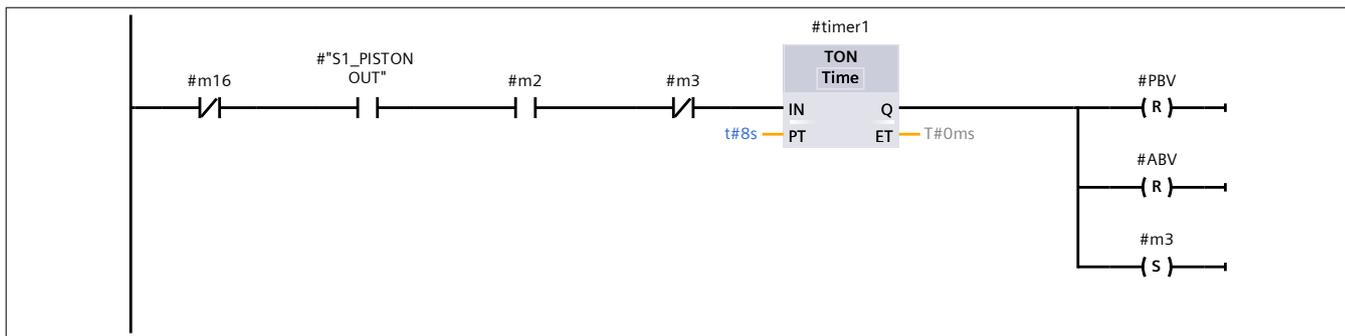
Set a memory for Leak Test



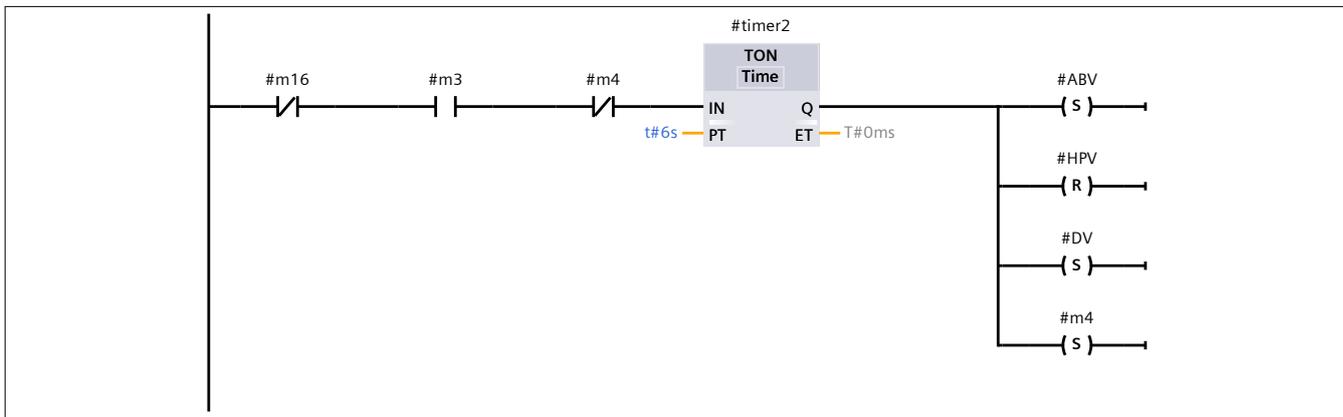
Network 2: Leak_Cylinder MOVE OUT



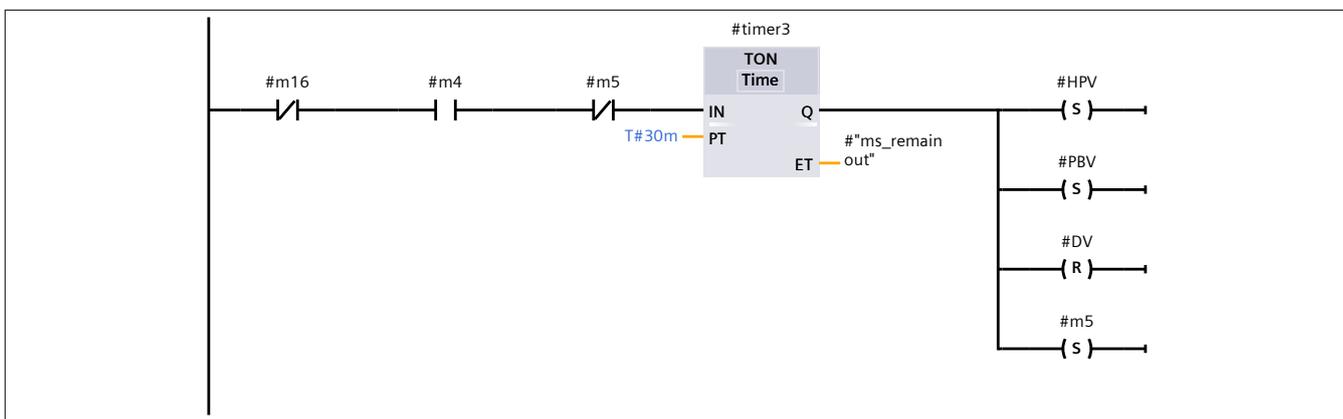
Network 3: Leak_Cylinder Under the Pressure Time when is out



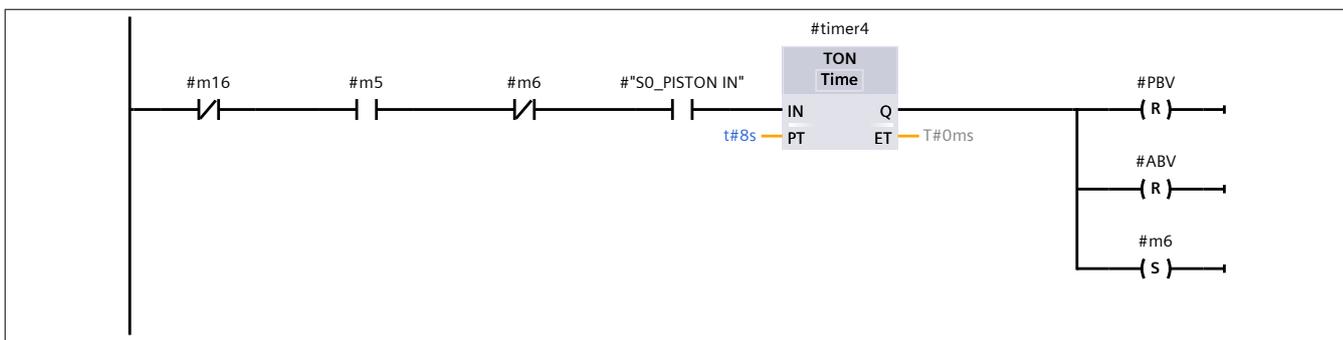
Network 4: Leak_Disconnecting The Supply Source when Cylinder is out



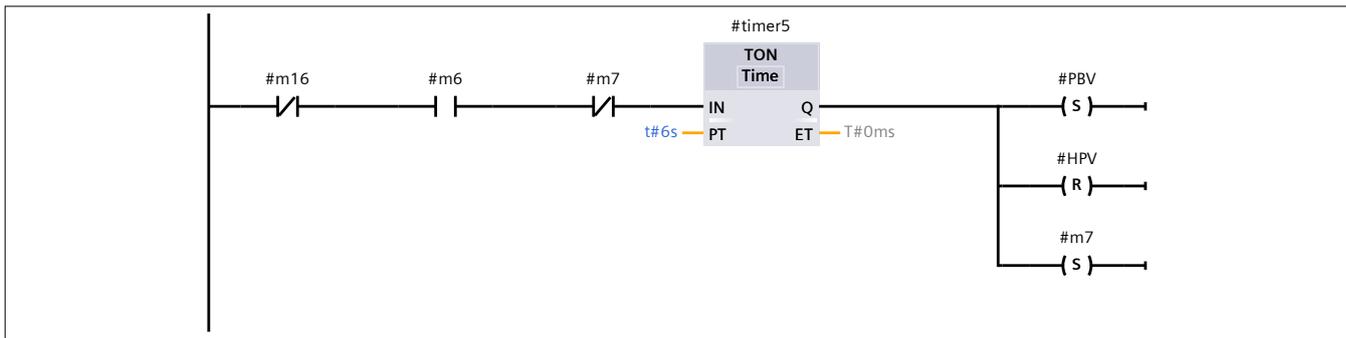
Network 5: Leak_Piston remains OUT For 30 Minutes



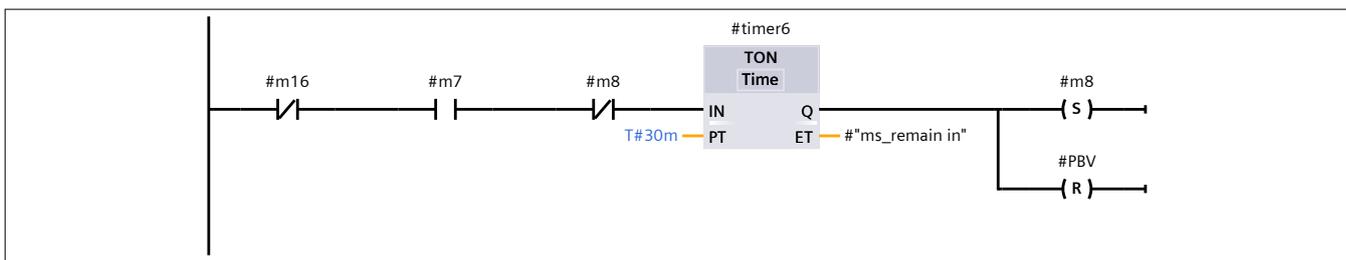
Network 6: Leak_Cylinder Move IN & Under the Pressure Time



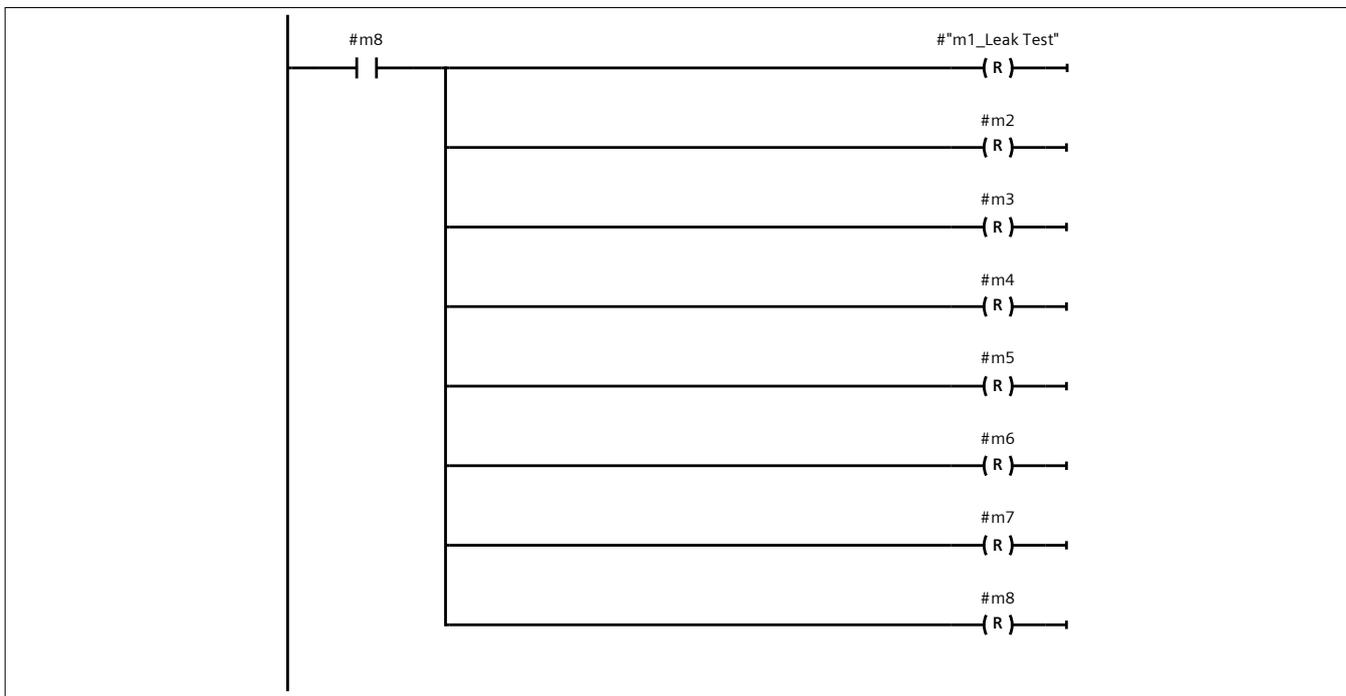
Network 7: Leak_Disconnecting The Supply Source when is Inside



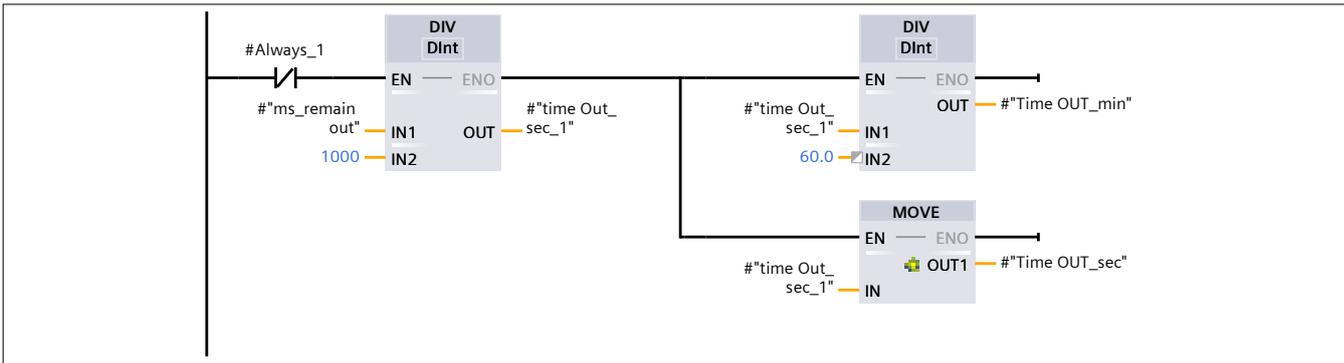
Network 8: Leak_Piston remains Inside for 30 Minutes



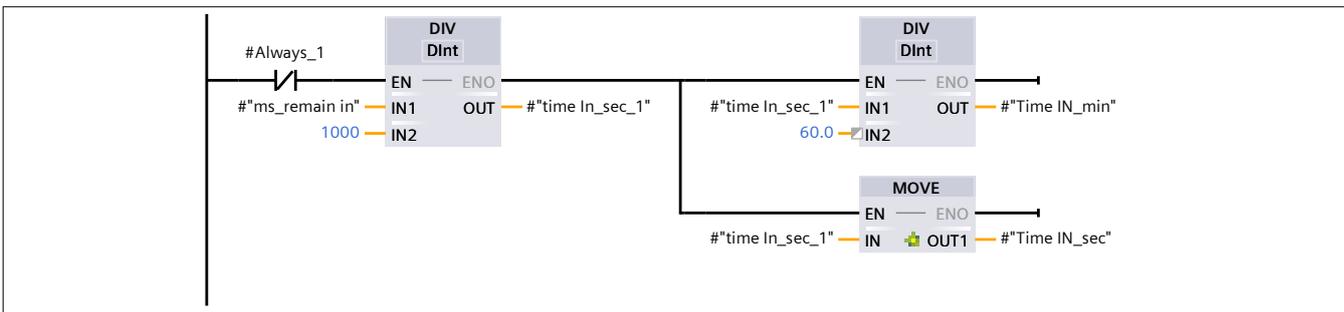
Network 9: Leak_Go back to first Network



Network 10: Leak_Convert ms-remain OUT Time to second and minutes

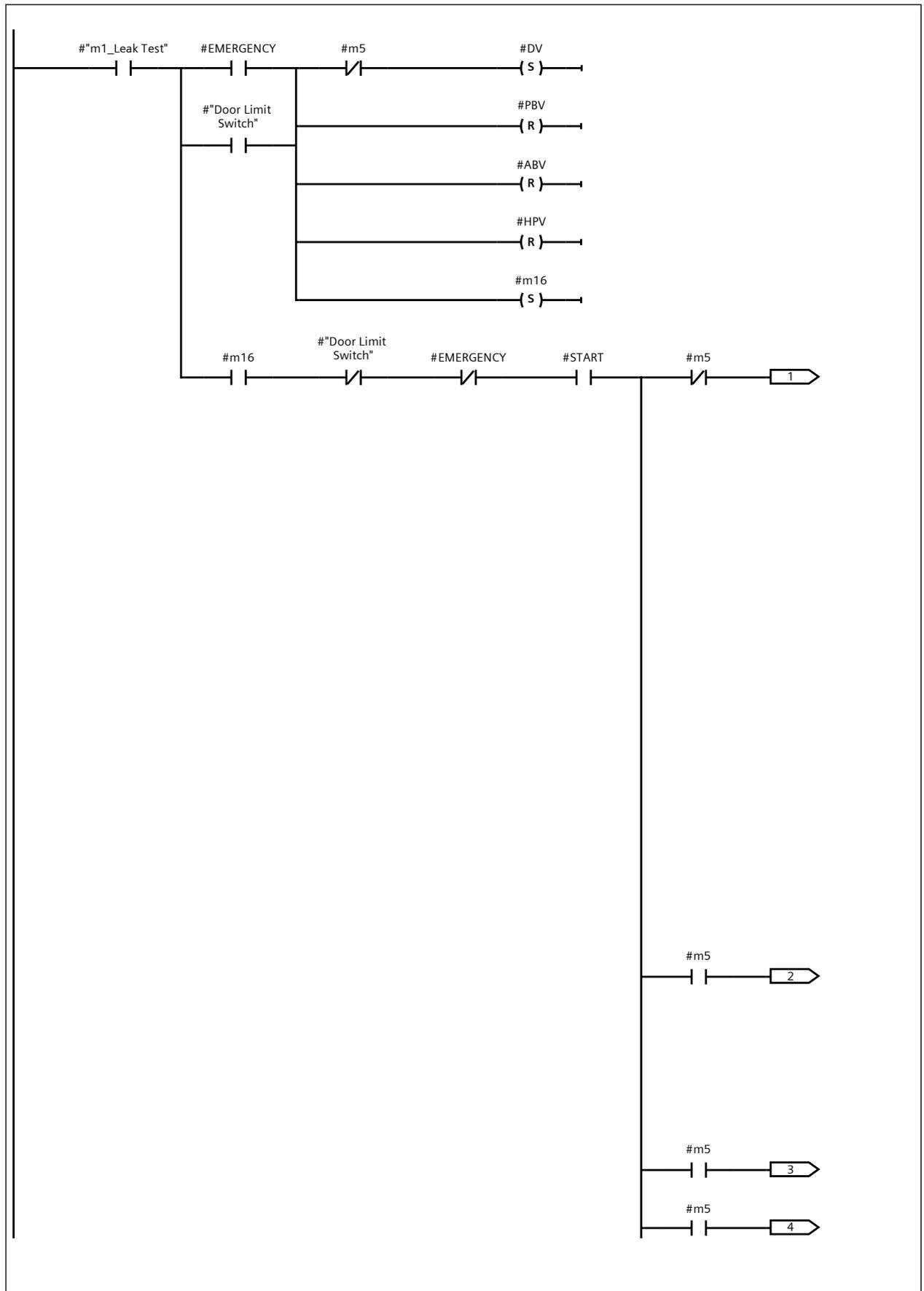


Network 11: Leak_Convert remain IN Time- ms to second and minutes



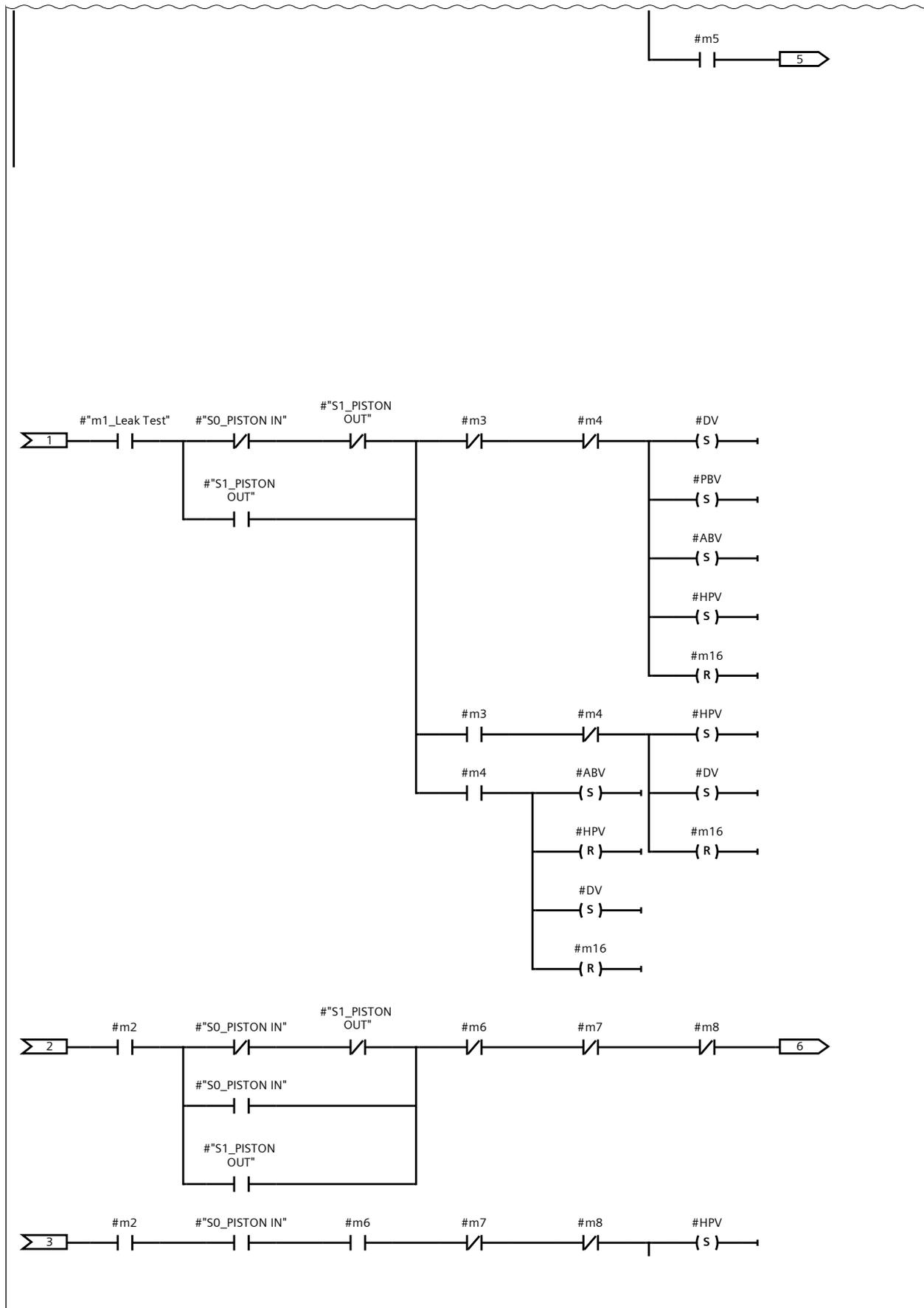
Network 12: Leak_Emergency Leak Test

Network 12: Leak_Emergency Leak Test (1.1 / 4.1)



Network 12: Leak_Emergency Leak Test (2.1 / 4.1)

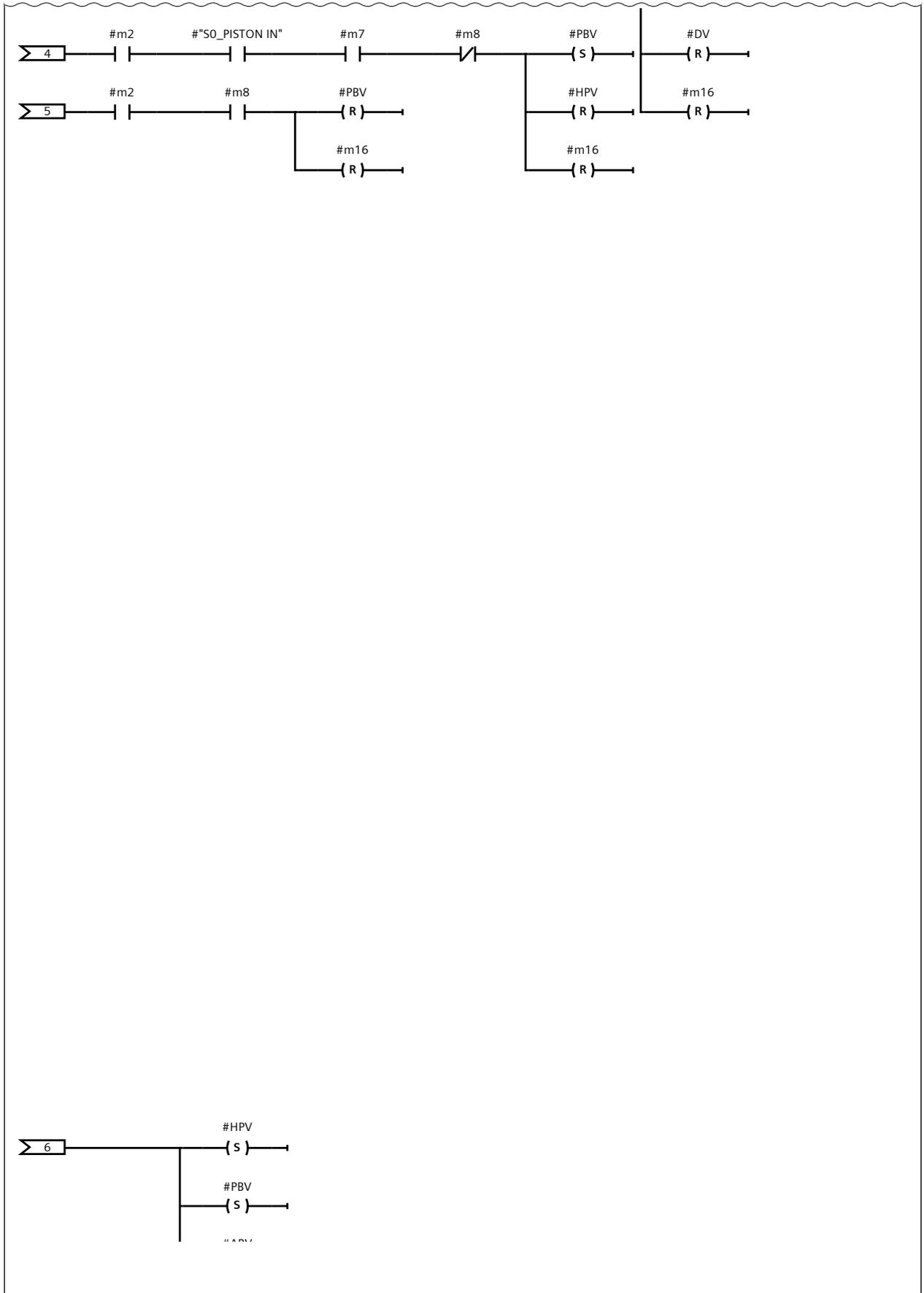
1.1 (Page1 - 10)



3.1 (Page1 - 12)

Network 12: Leak_Emergency Leak Test (3.1 / 4.1)

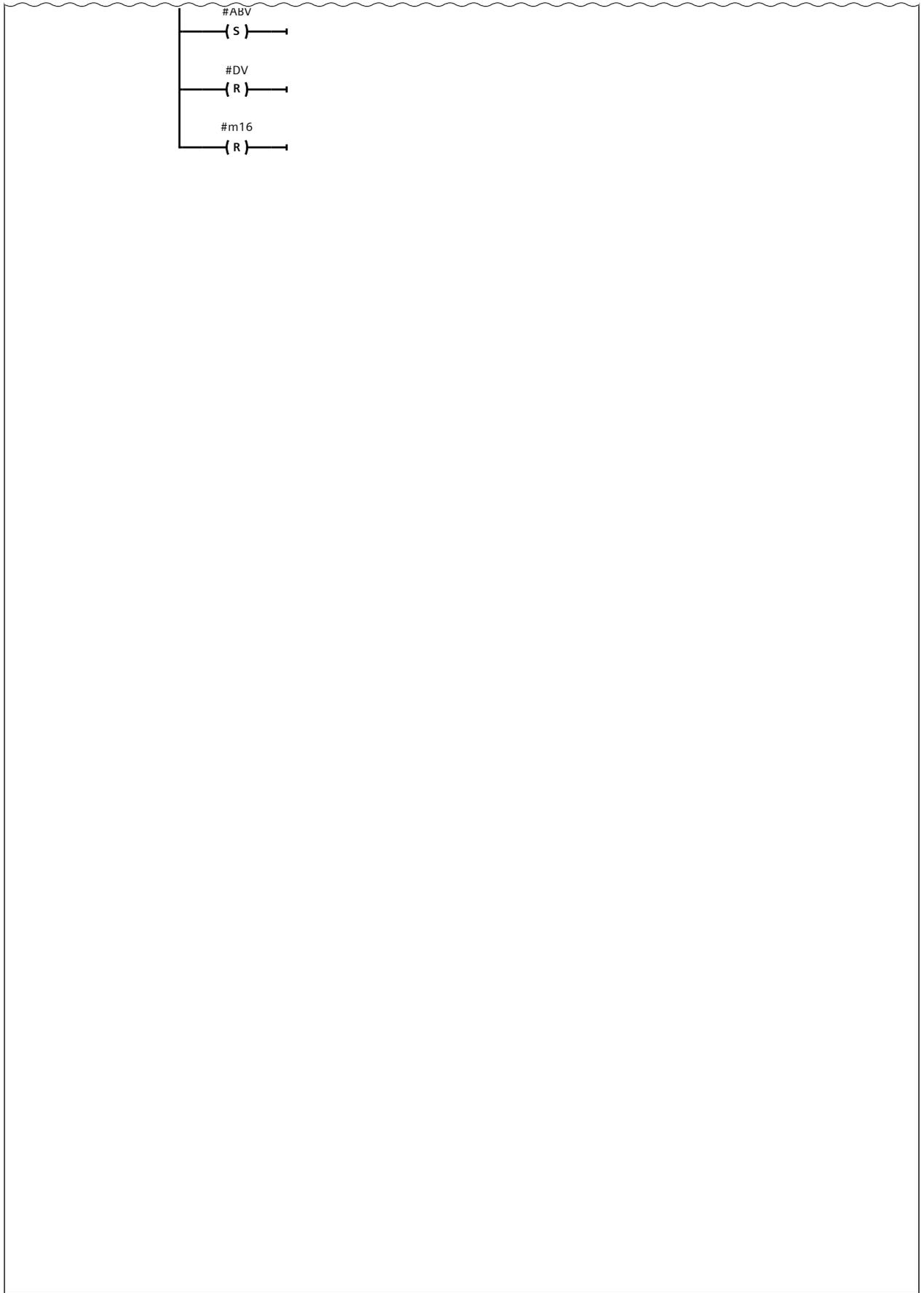
2.1 (Page1 - 11)



4.1 (Page1 - 13)

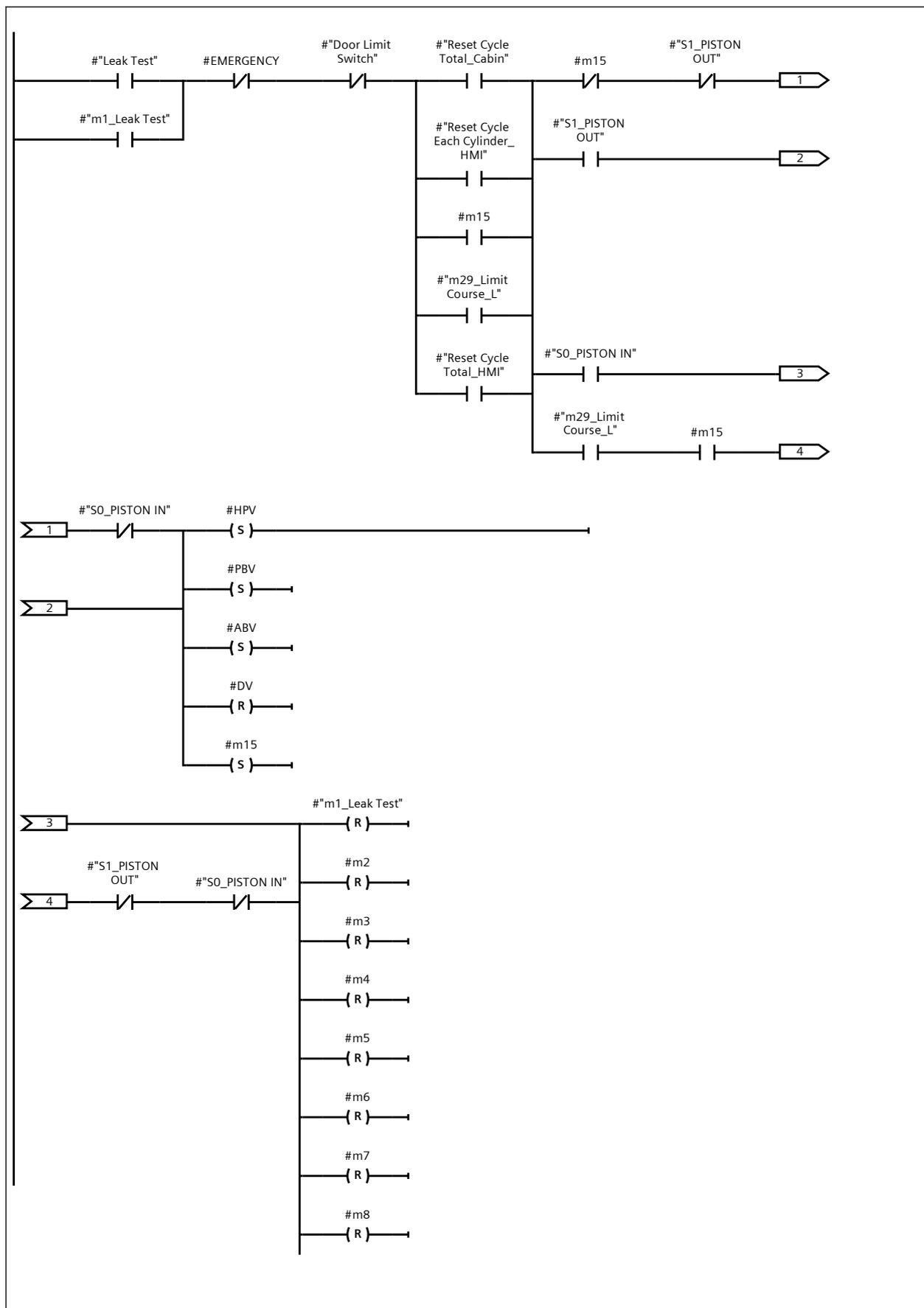
Network 12: Leak_Emergency Leak Test (4.1 / 4.1)

3.1 (Page1 - 12)



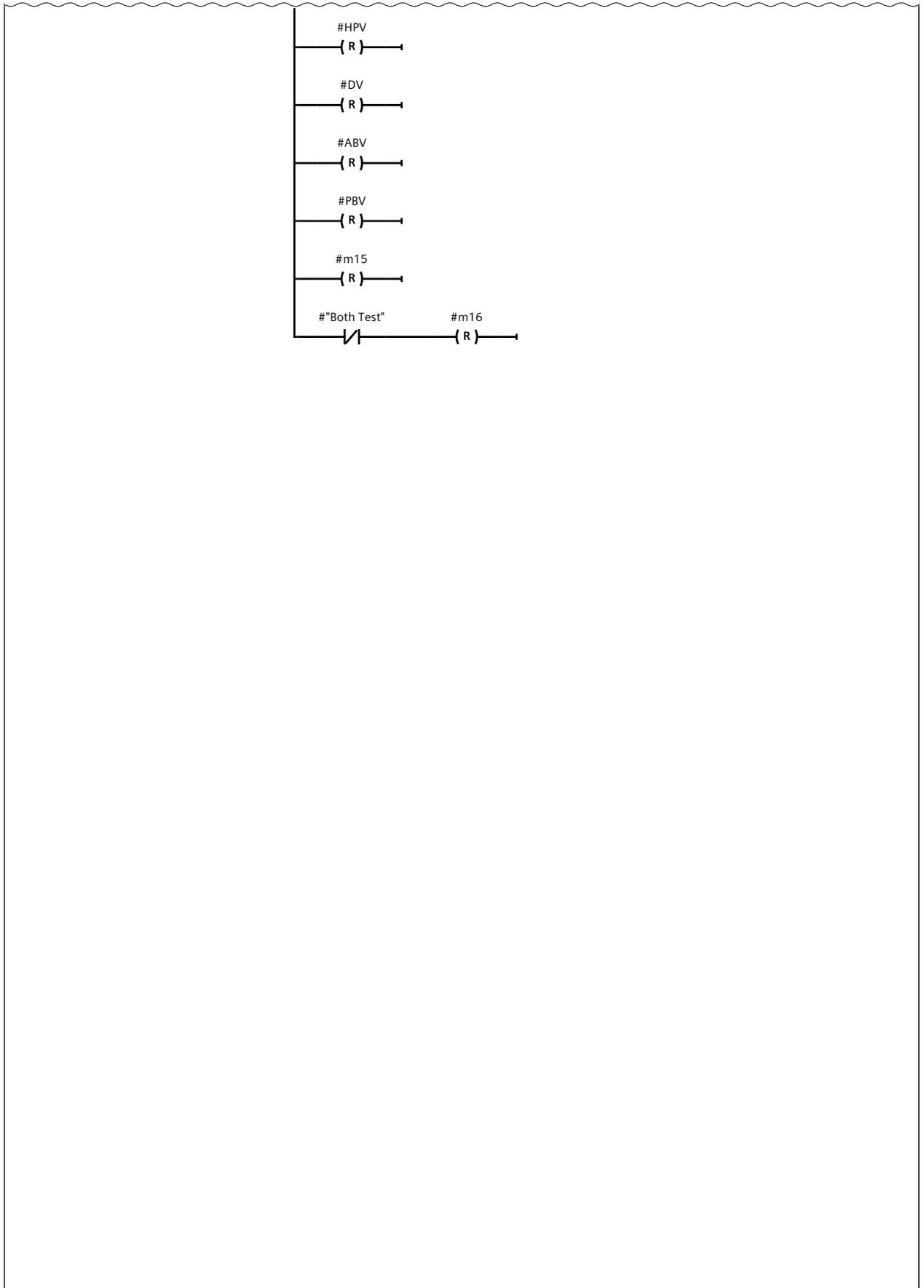
Network 13: Leak_Reset Process in Leak Test

Network 13: Leak_Reset Process in Leak Test (1.1 / 2.1)



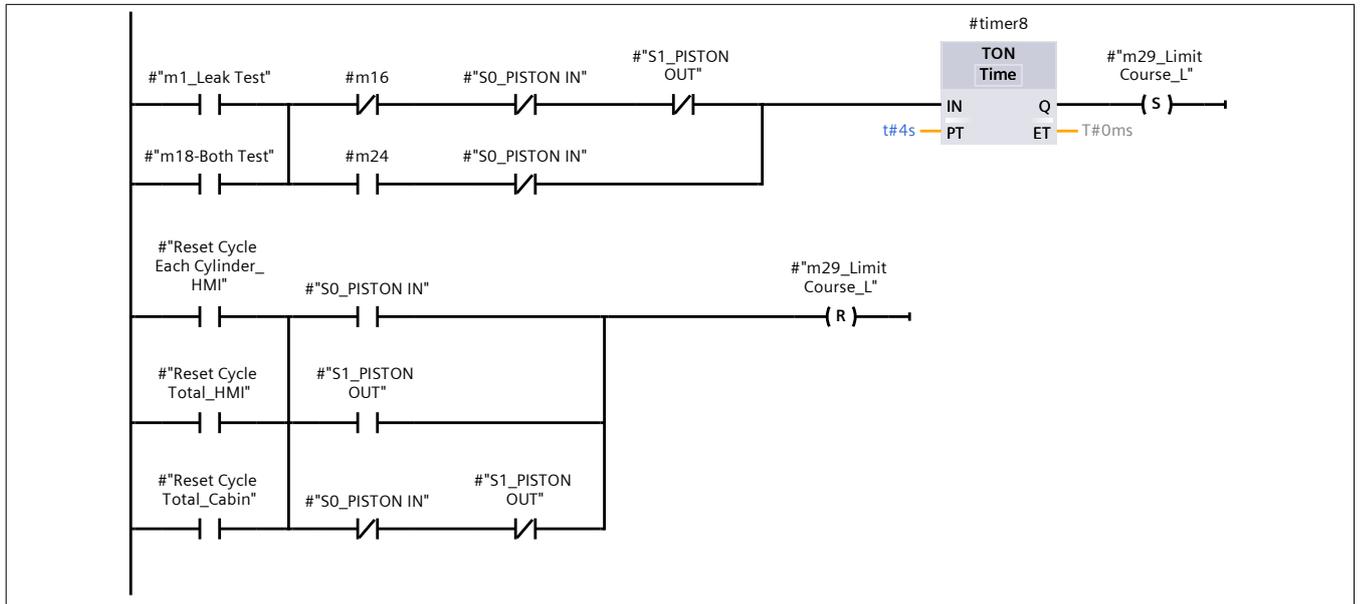
Network 13: Leak_Reset Process in Leak Test (2.1 / 2.1)

1.1 (Page1 - 15)

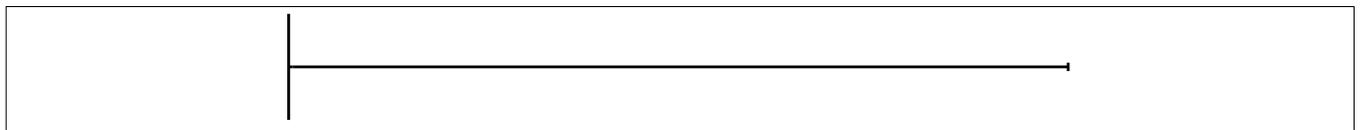


Network 14: Leak_Limitation Time Course

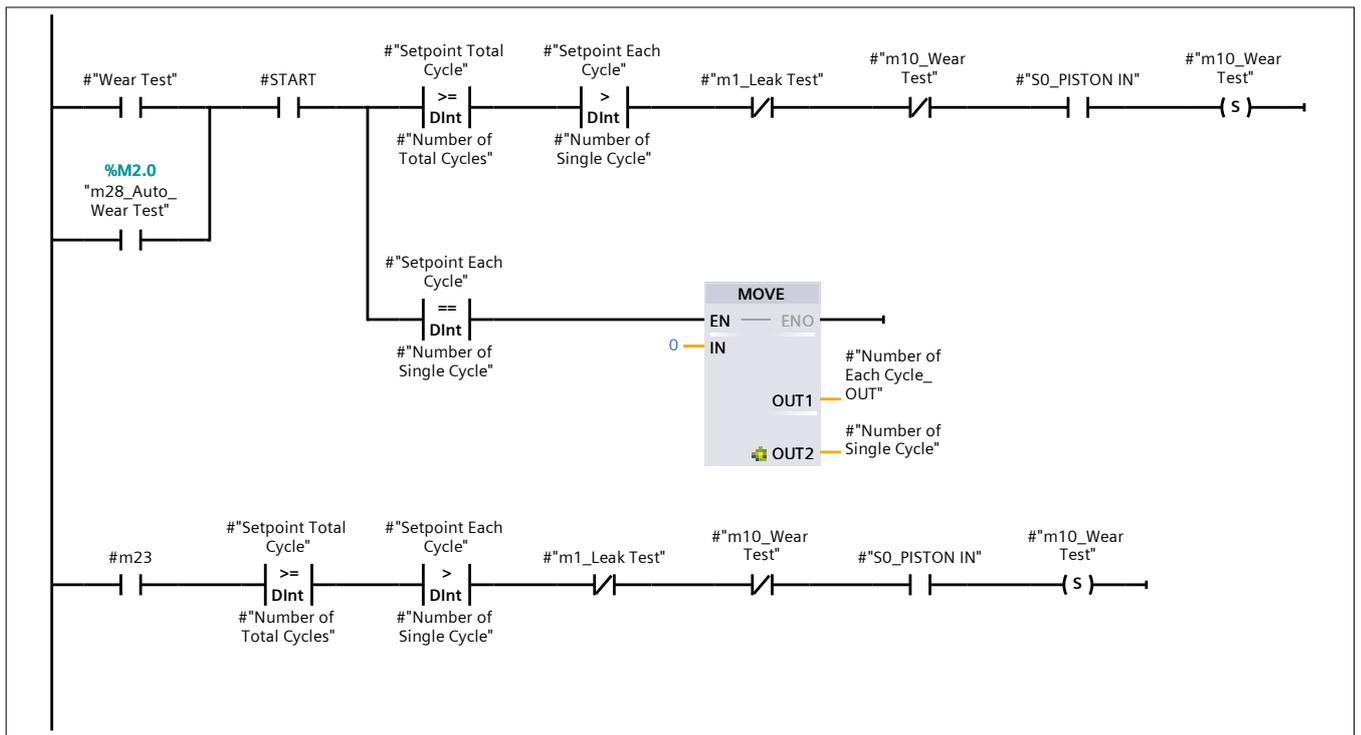
Distinguish the broken cylinders in the middle of the course or any sensor error_Set a memory & Reset the memory



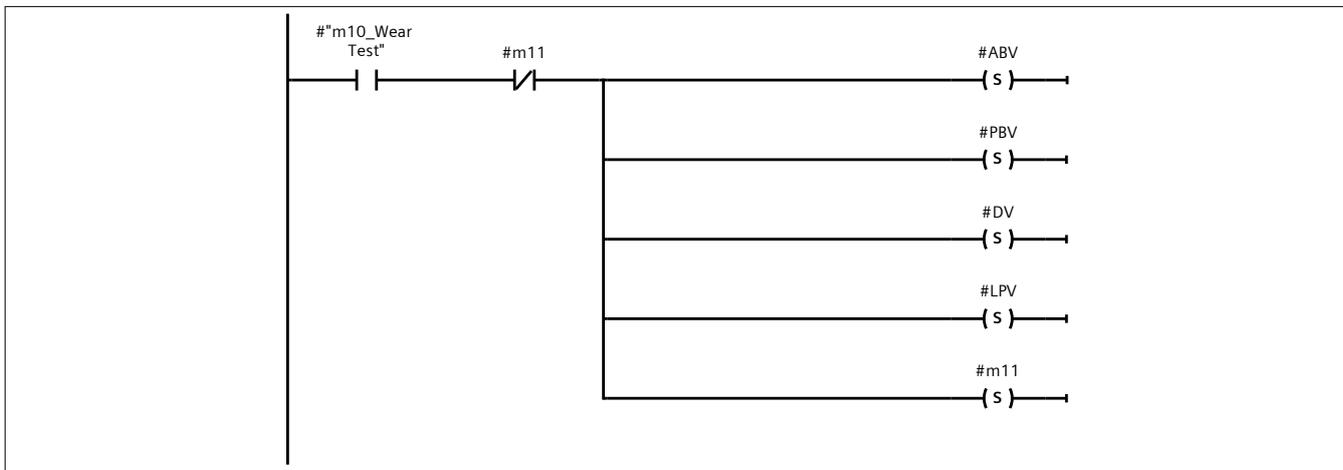
Network 15: -----



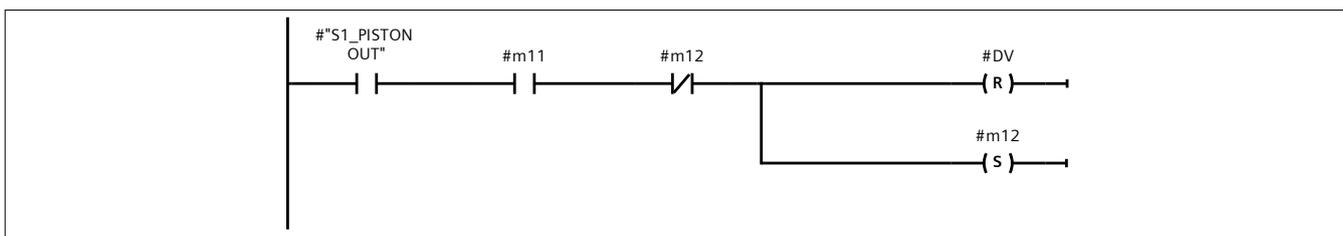
Network 16: Wear_Go to Wear Test Mode



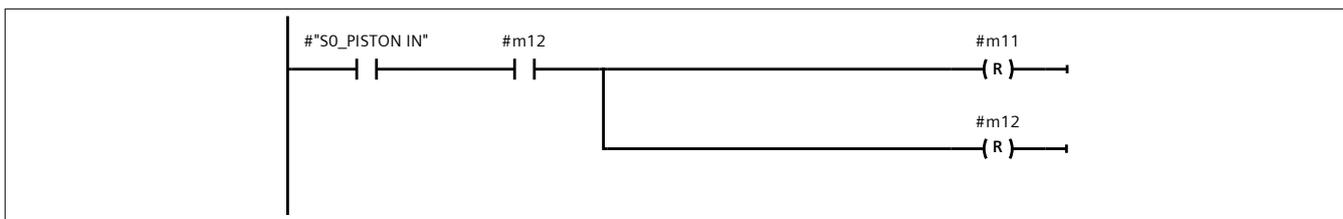
Network 17: Wear_Cylinder Move OUT



Network 18: Wear_Cylinder Move IN

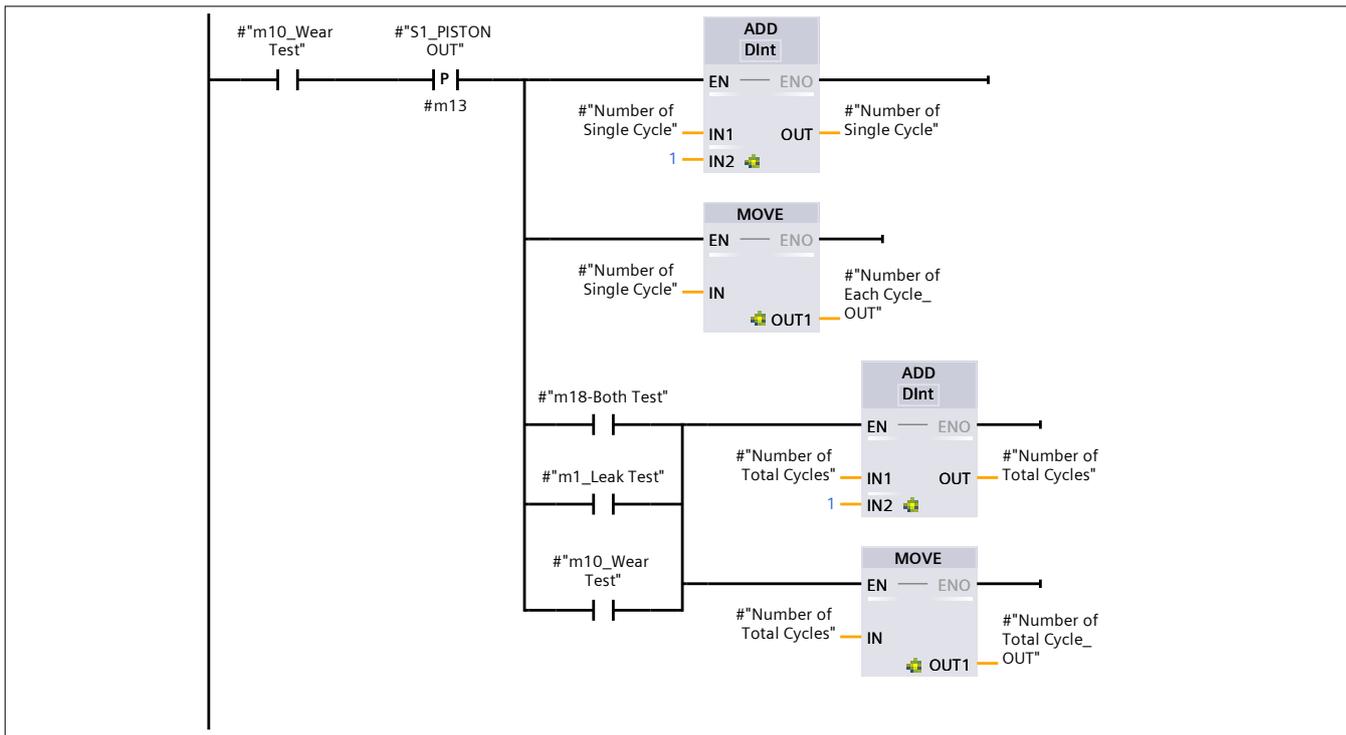


Network 19: Wear_Reapet Wear Cycle Test



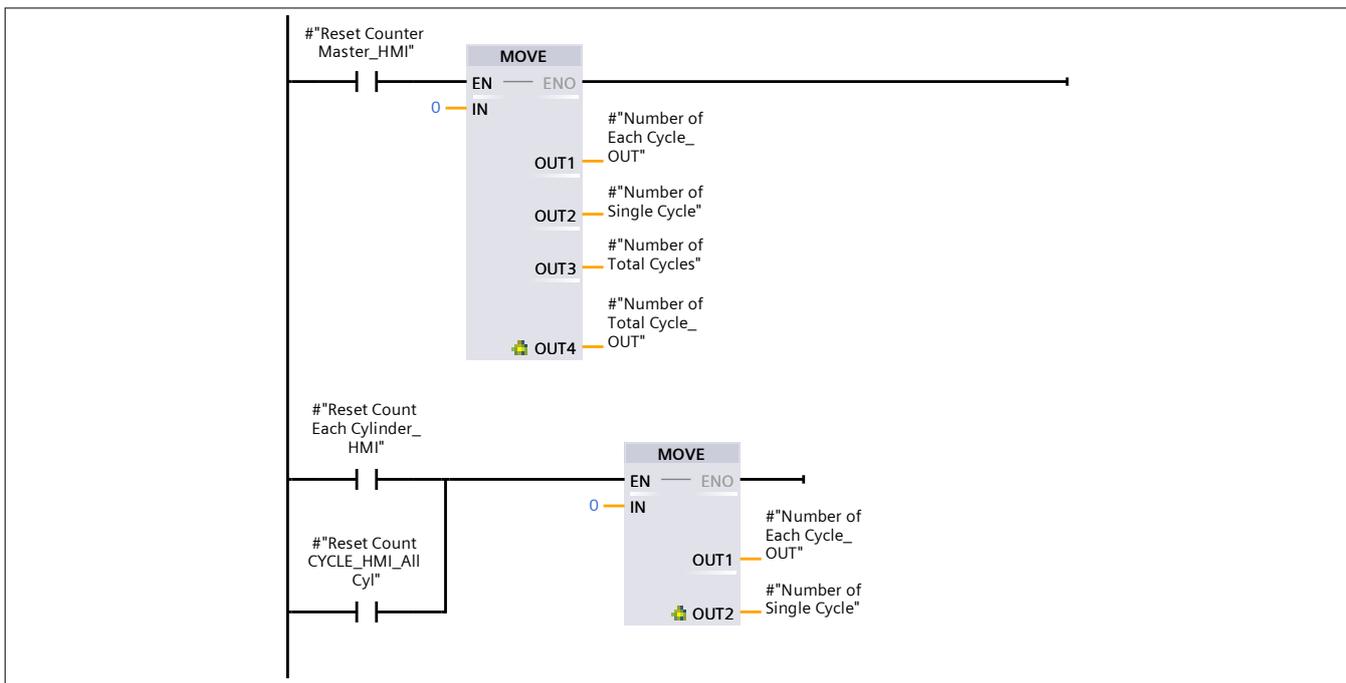
Network 20: Wear_Count Wear cycles

Creating Counter to count more than ordinary counters

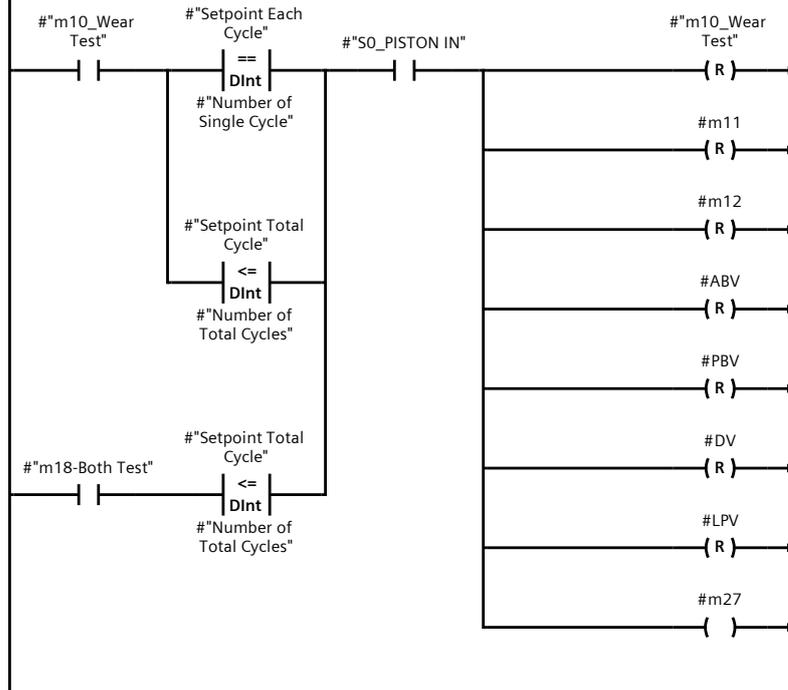


Network 21: Wear_Reset Counter

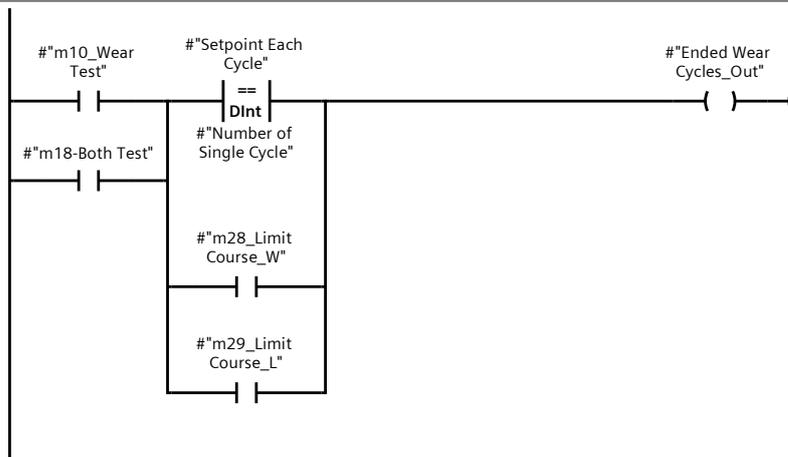
Reset Counters



Network 22: Wear_End Cycle

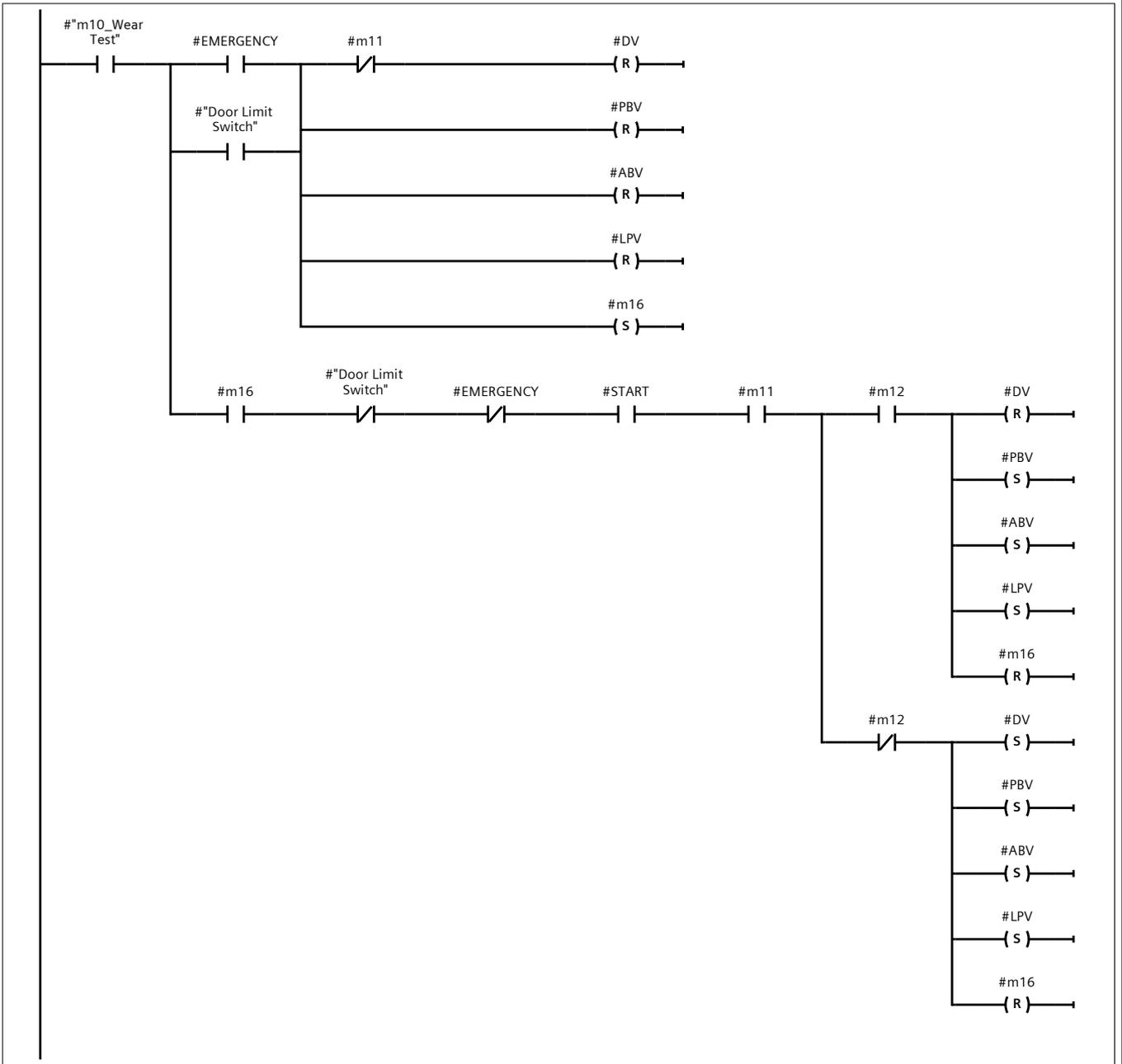


Network 23: Wear_Set a Memory for "Wear Cycles Ended"



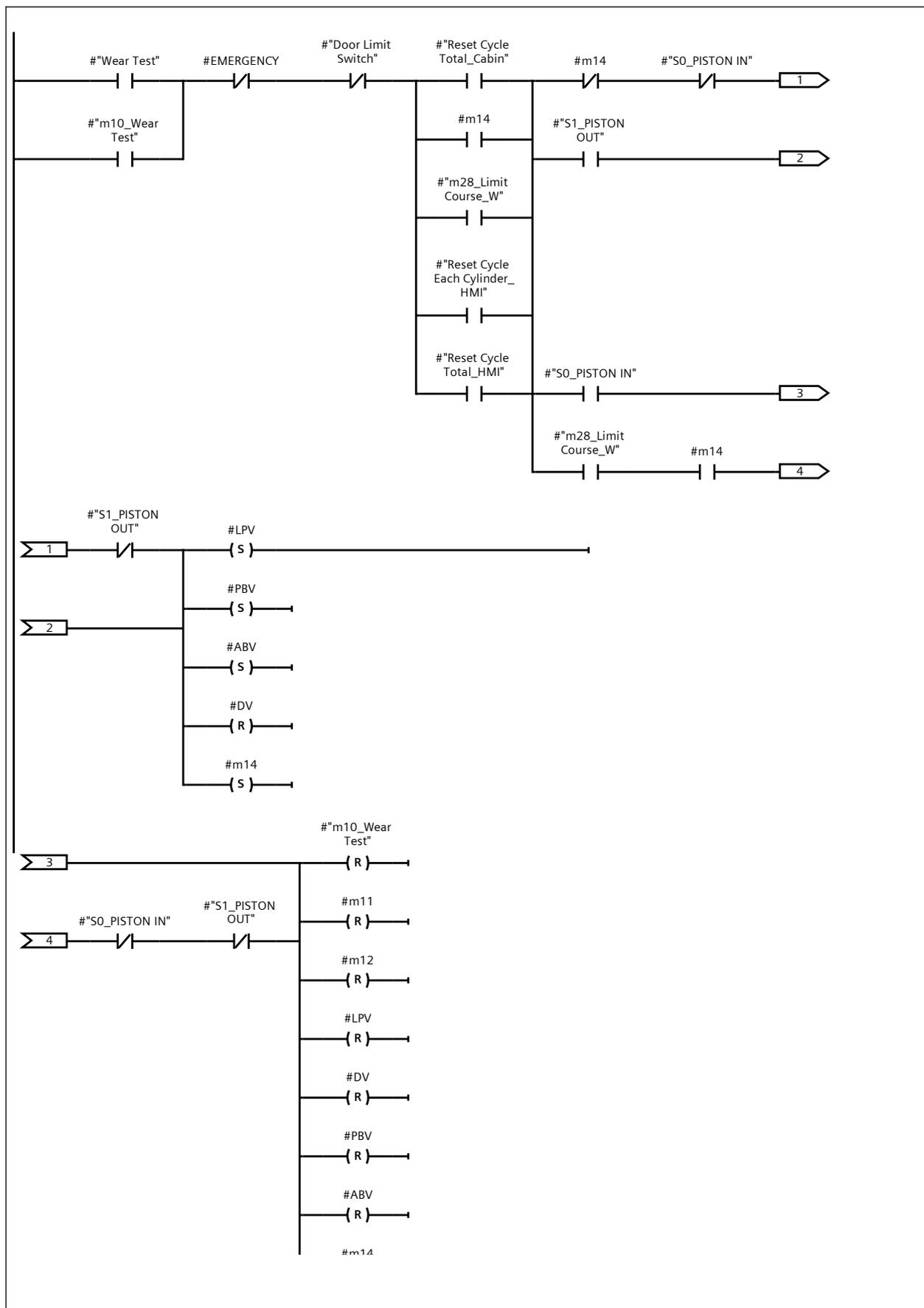
Network 24: Wear_Emergency Wear Test

Urgent Emergency to Stop the Test Bench and Cylinders in the exact current positions



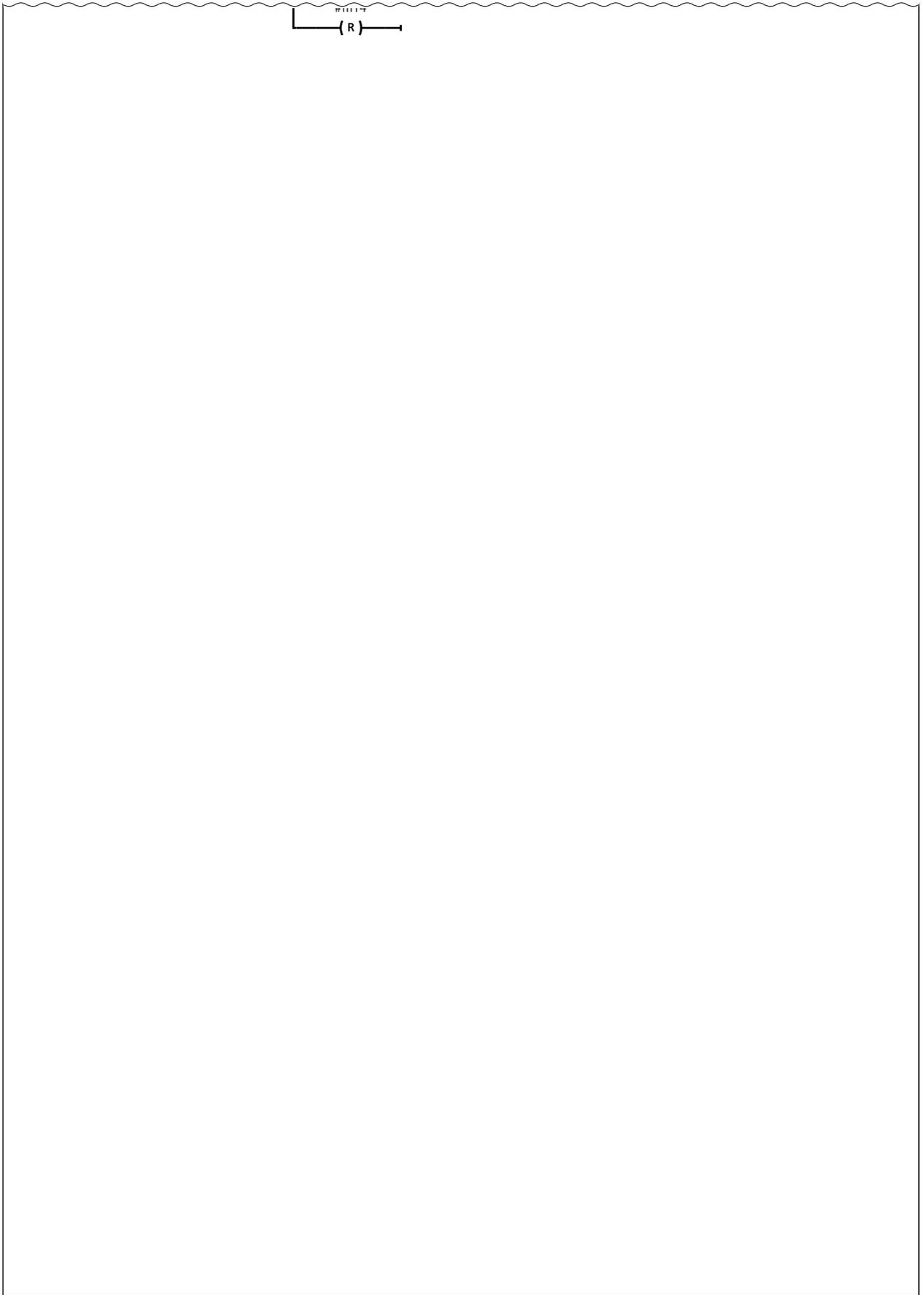
Network 25: Wear_Reset Process Wear Test

Network 25: Wear_Reset Process Wear Test (1.1 / 2.1)



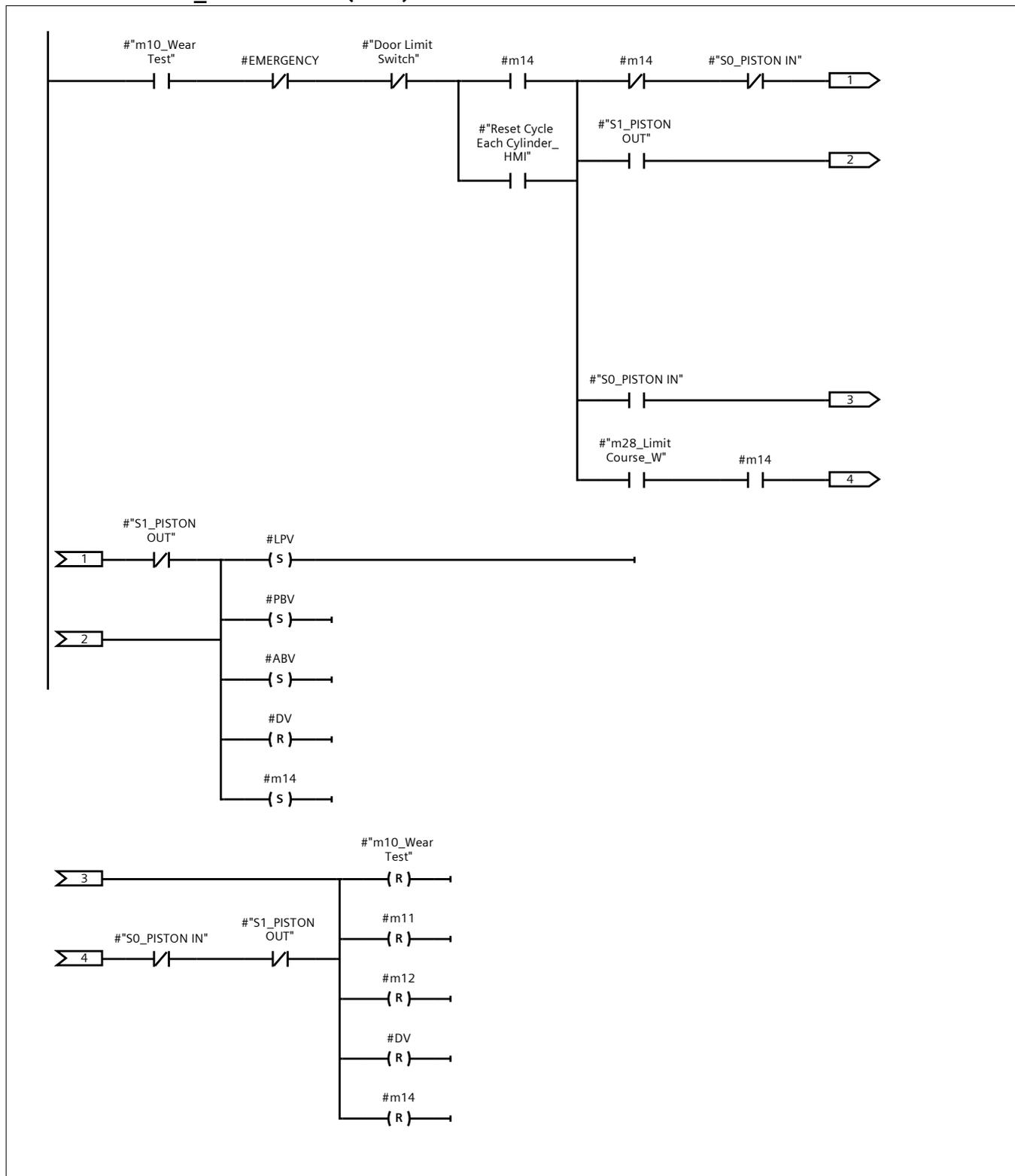
Network 25: Wear_Reset Process Wear Test (2.1 / 2.1)

1.1 (Page1 - 22)



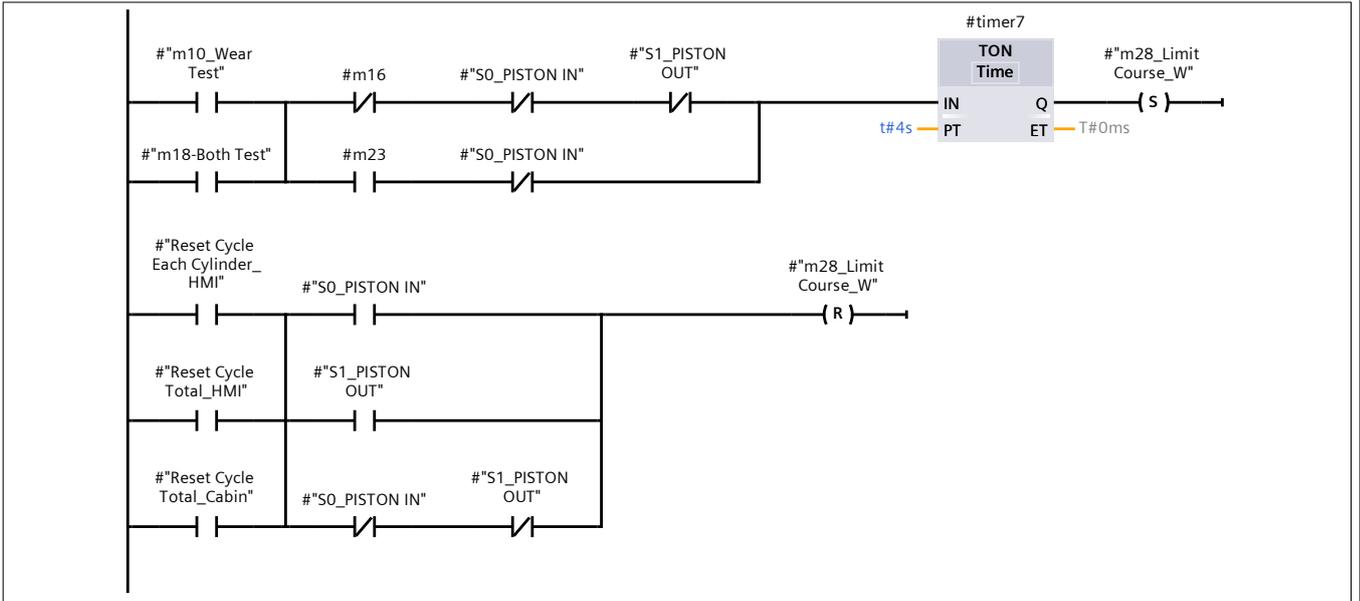
Network 26: Wear_Reset Process(HMII) Wear Test

Network 26: Wear_Reset Process(HMII) Wear Test



Network 27: Wear_Limitation Time Course

Distinguish the broken cylinders in the middle of the course or any sensor error_Set a memory & Reset the memory

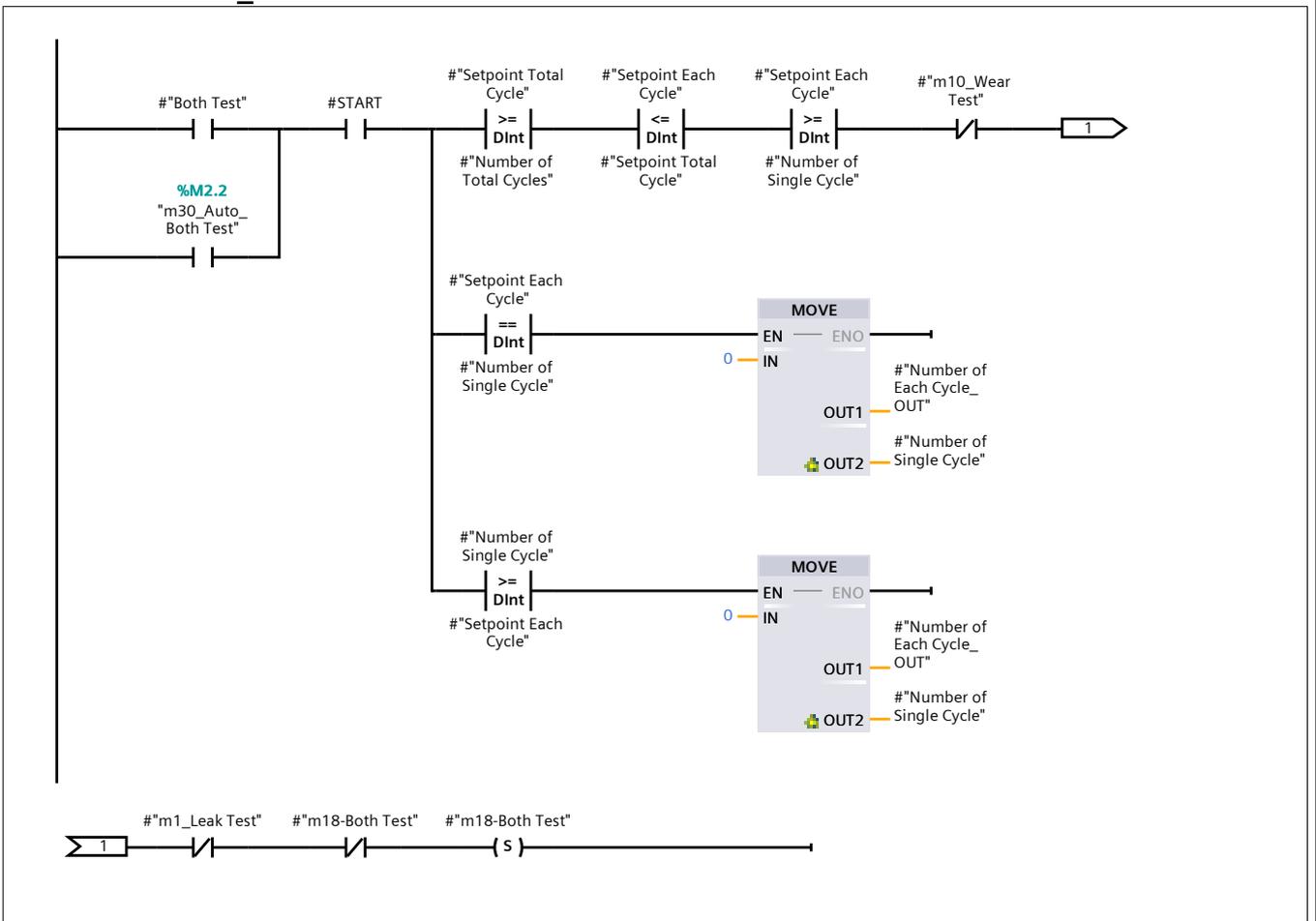


Network 28: -----

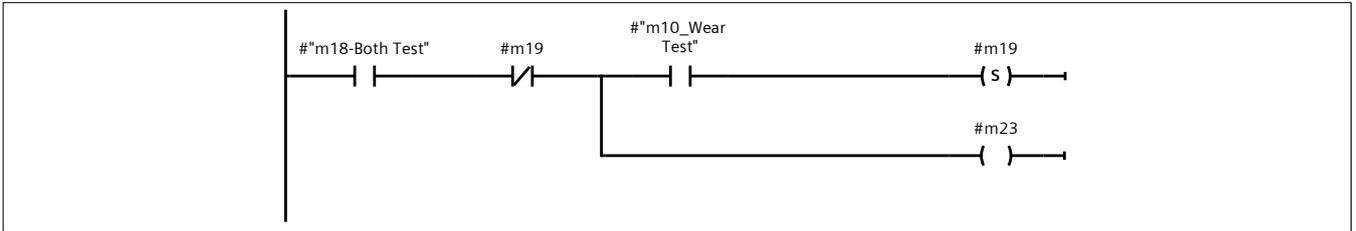
Network 29: Both_Go to Both Test Mode

Arranging to do BOTH tests successively one after the other one

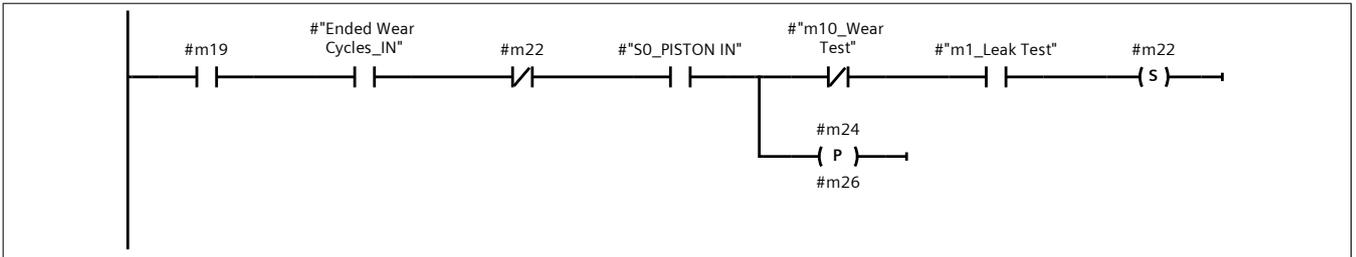
Network 29: Both_Go to Both Test Mode



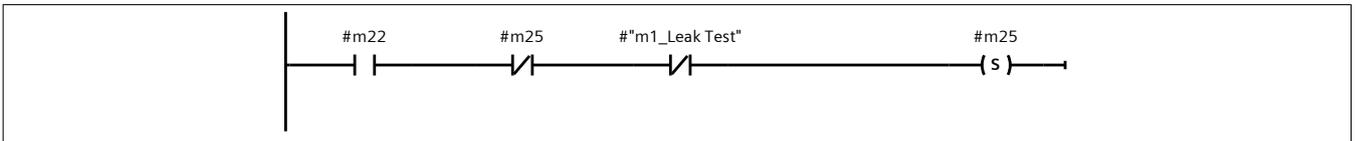
Network 30: Both_Start Wear Test



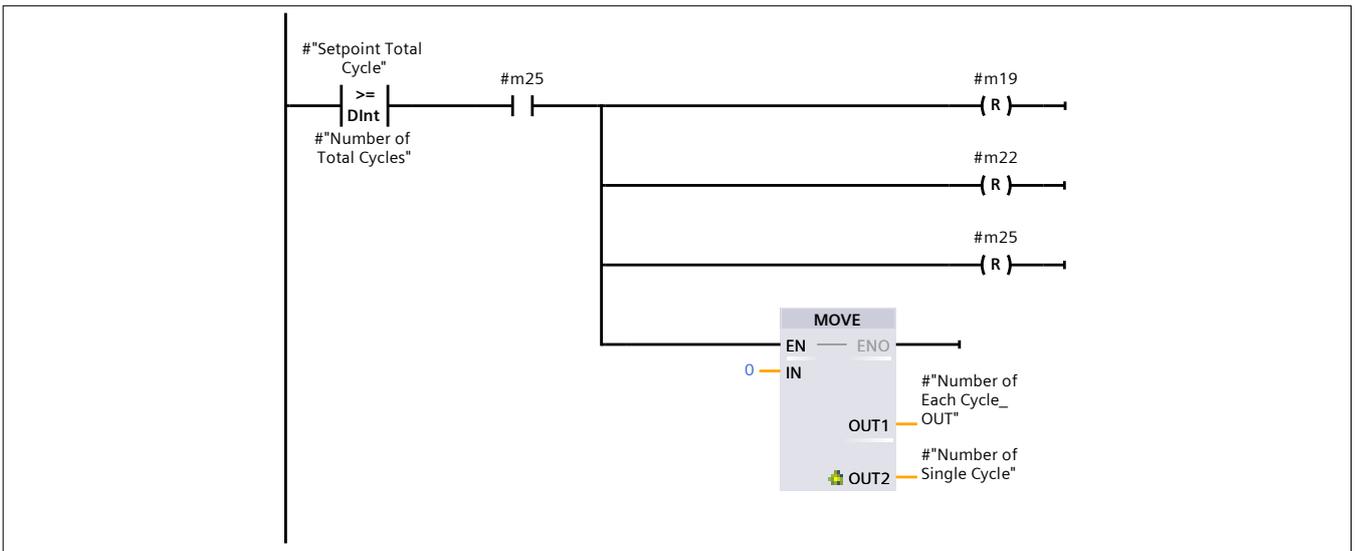
Network 31: Both_Start Leak Test



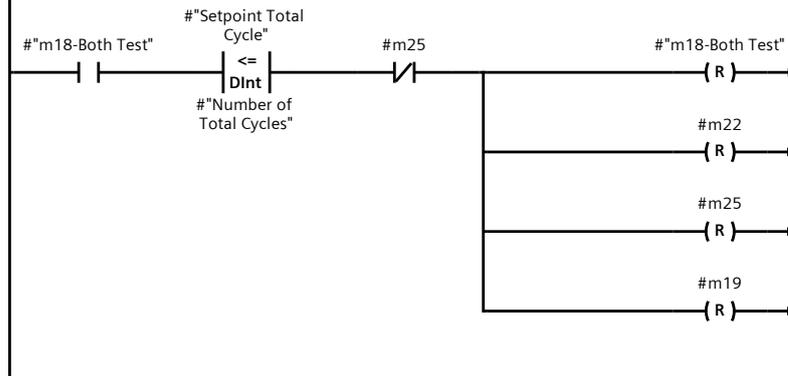
Network 32: Both_End Leak Test



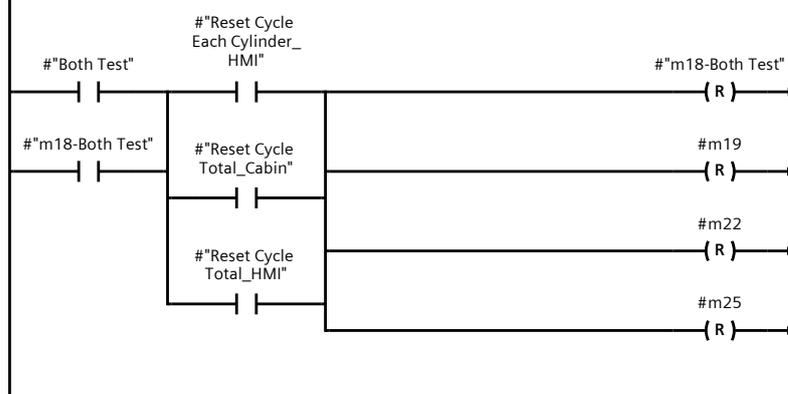
Network 33: Both_Repeat Cycle in Both Mode



Network 34: Both_End Cycles in Both Mode



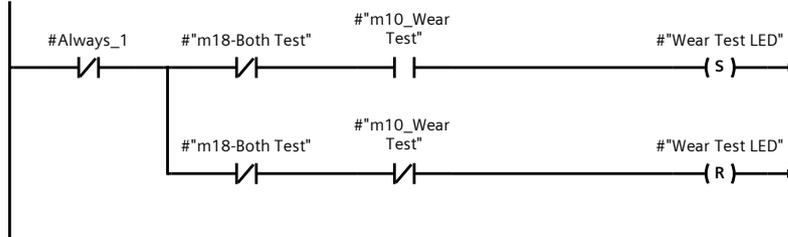
Network 35: Both_Reset Process in Both Mode



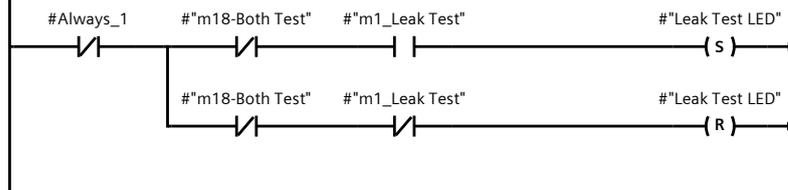
Network 36: -----



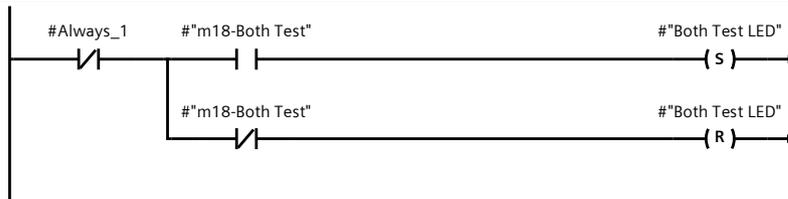
Network 37: Wear Test LED



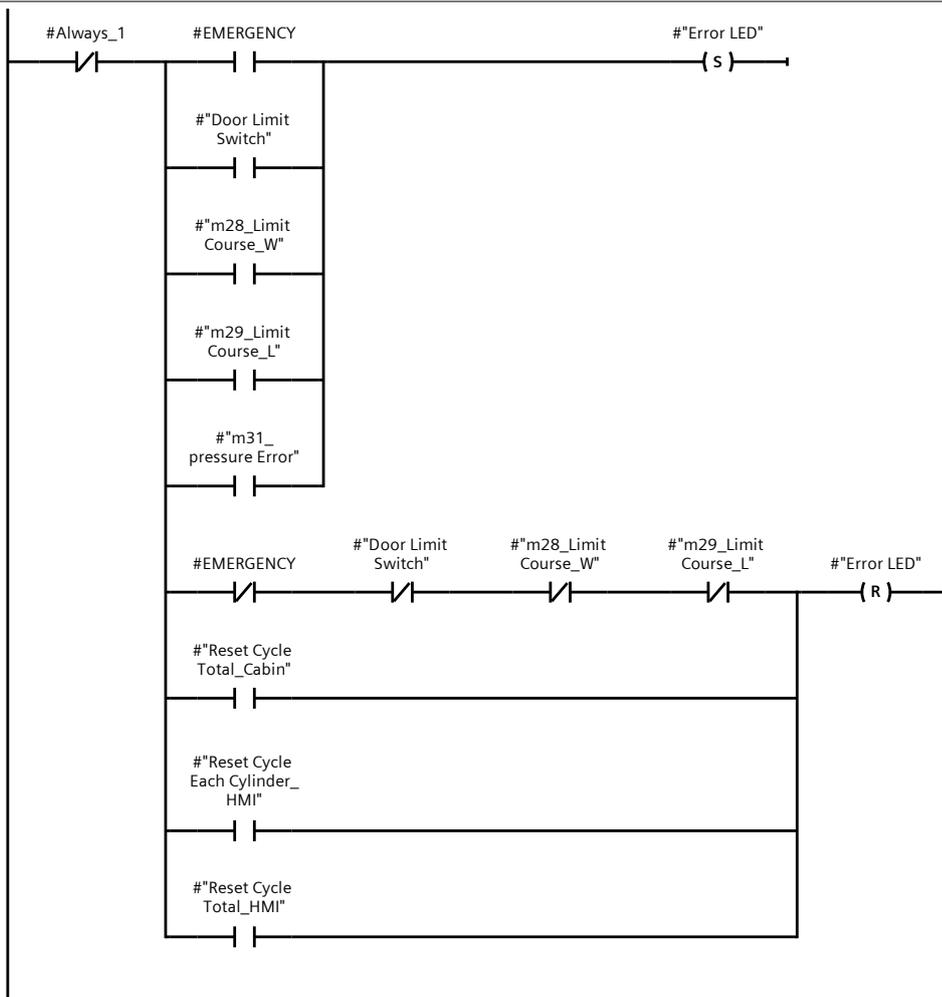
Network 38: Leak Test LED



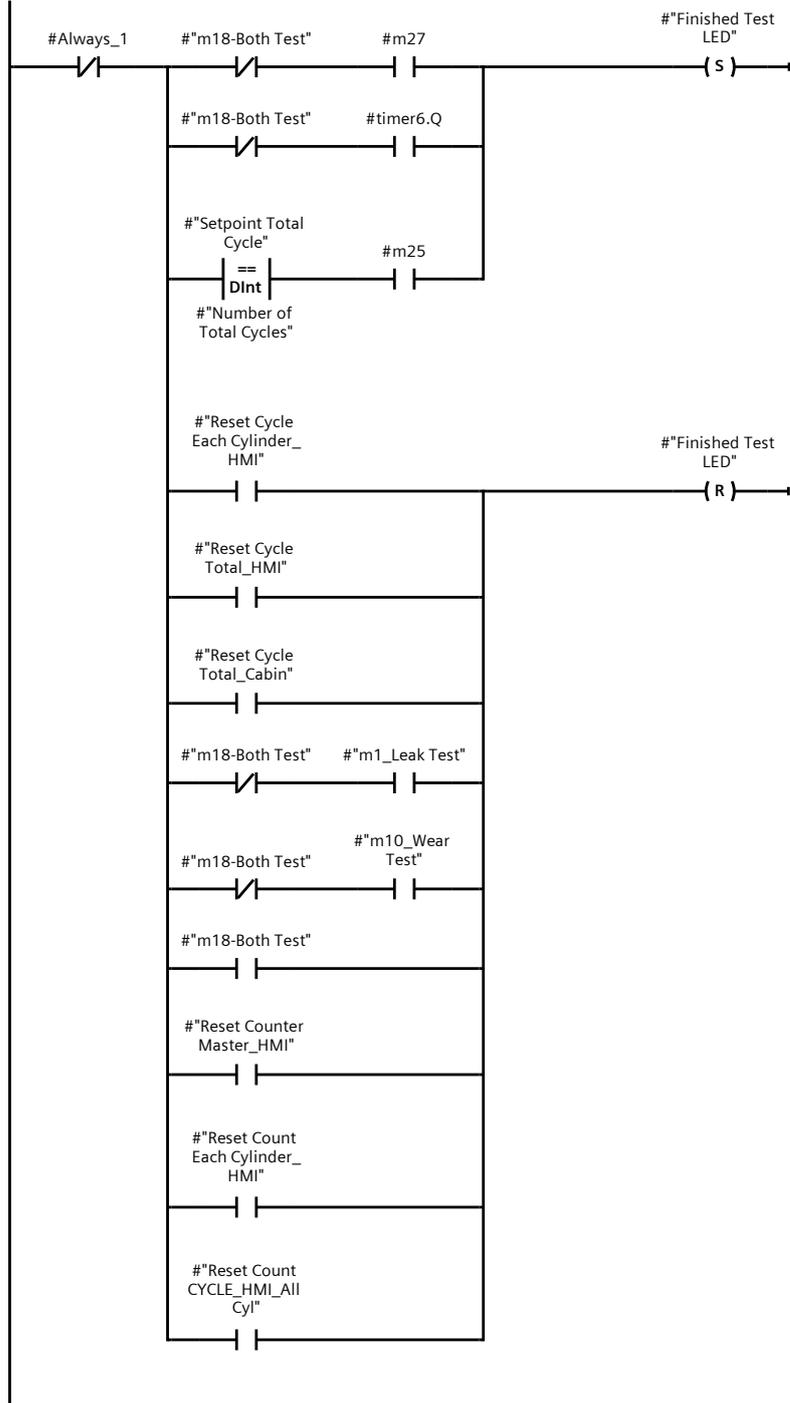
Network 39: Both Test LED



Network 40: Error LED



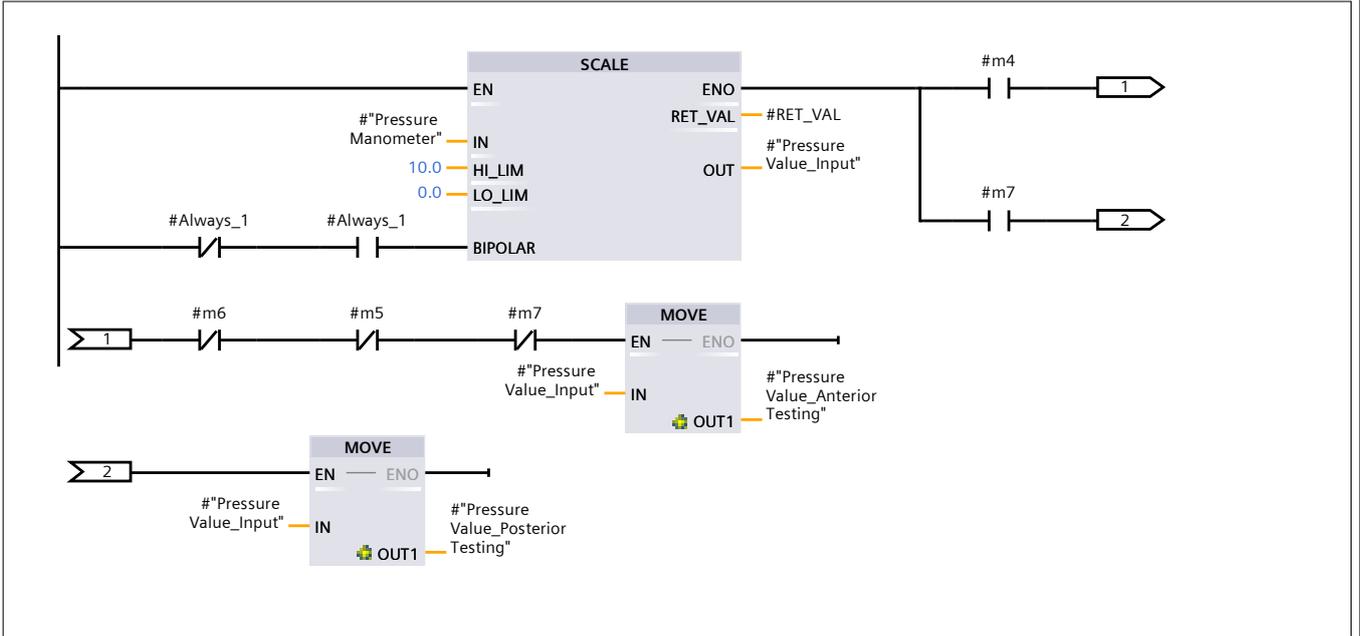
Network 41: Finished Test LED



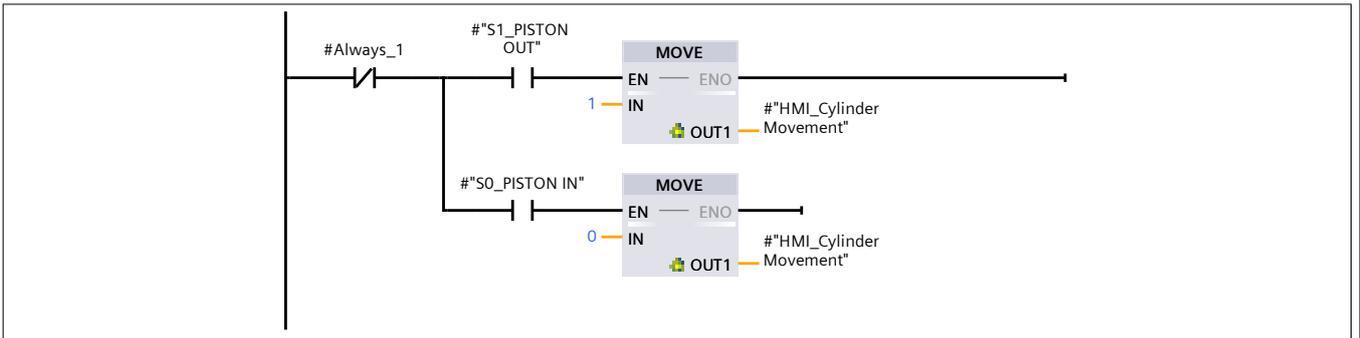
Network 42: -----

Network 43: Scale Manometer

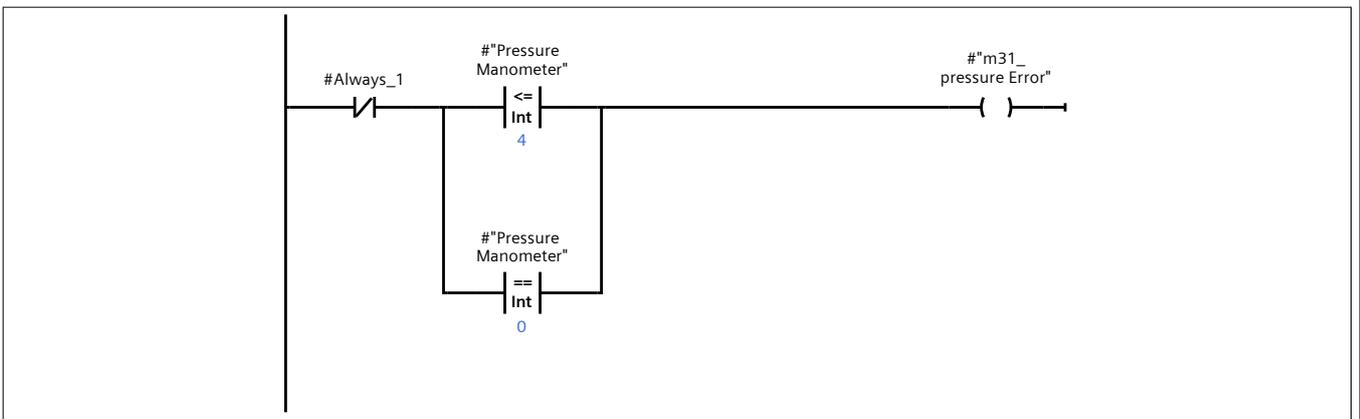
Network 43: Scale Manometer



Network 44: HMI Screen Cylinder



Network 45: Error if the input pressure is less than 4 bar



The end of Test Bench Program Block

Automation for All Cyl_22 / PLC_6 [CPU 1512C-1 PN] / Program blocks

Selector Test Mode_HMI [FB2]

Selector Test Mode_HMI Properties

General

Name	Selector Test Mode_HMI	Number	2	Type	FB
Language	LAD	Numbering	Automatic		

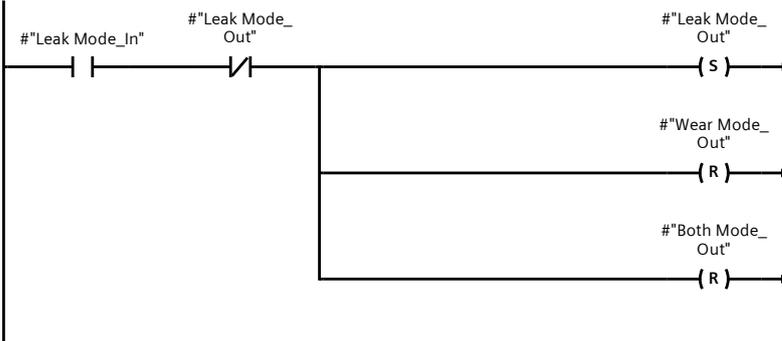
Information

Title	3-Positiom Selector Test Mode on HMI	Author		Comment	
Family		Version	0.1	User-defined ID	

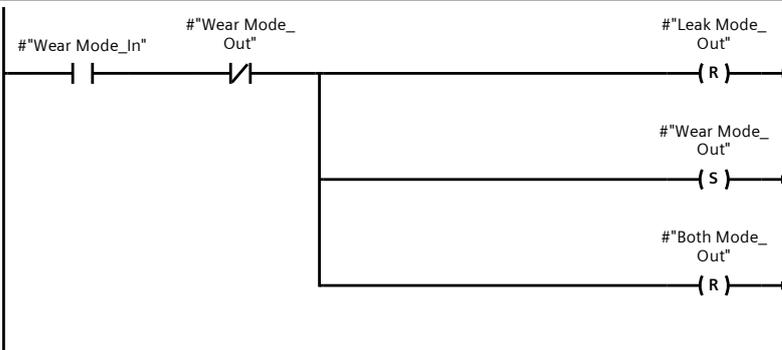
Selector Test Mode_HMI

Name	Data type	Default value	Retain	Access-ible from HMI/OP C UA	Wri-ta-ble from HM I/O PC UA	Visible in HMI engi-neer-ing	Set-point	Super-vision	Comment
▼ Input									
Wear Mode_In	Bool	false	Non-retain	True	True	True	False		
Both Mode_In	Bool	false	Non-retain	True	True	True	False		
Leak Mode_In	Bool	false	Non-retain	True	True	True	False		
Reset Auto Mode	Bool	false	Non-retain	True	True	True	False		
▼ Output									
Wear Mode_Out	Bool	false	Non-retain	False	False	False	False		
Both Mode_Out	Bool	false	Non-retain	False	False	False	False		
Leak Mode_Out	Bool	false	Non-retain	False	False	False	False		
InOut									
Static									
Temp									
Constant									

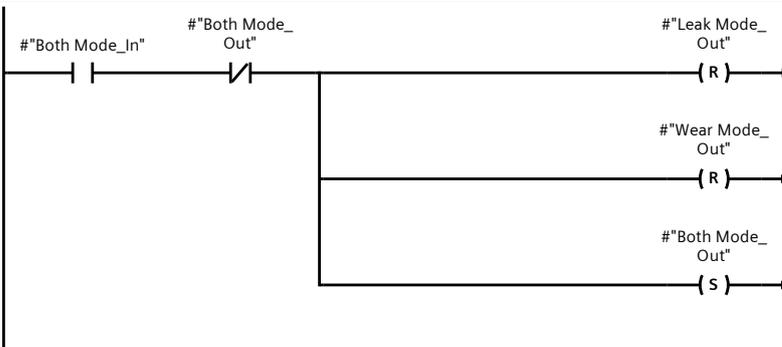
Network 1: Leak Mode Selection on HMI



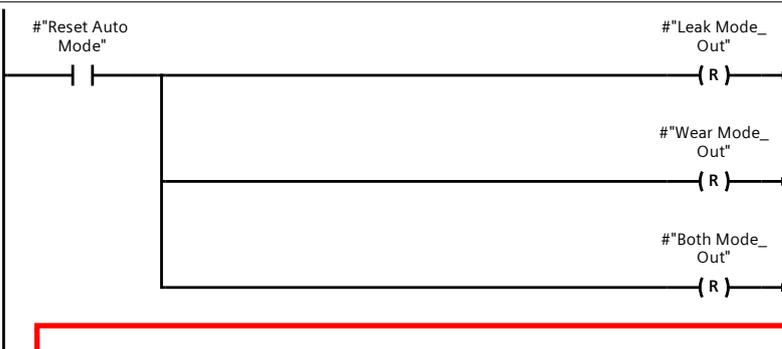
Network 2: Wear Mode Selection on HMI



Network 3: Both Mode Selection on HMI



Network 4: Selection REST Mode on HMI



The end of the Selector switch Block

Automation for All Cyl_22 / PLC_6 [CPU 1512C-1 PN] / Program blocks

Light around Test Modes Buttons [FC1]

Light around Test Modes Buttons Properties

General

Name	Light around Test Modes Buttons	Number	1	Type	FC
Language	LAD	Numbering	Automatic		

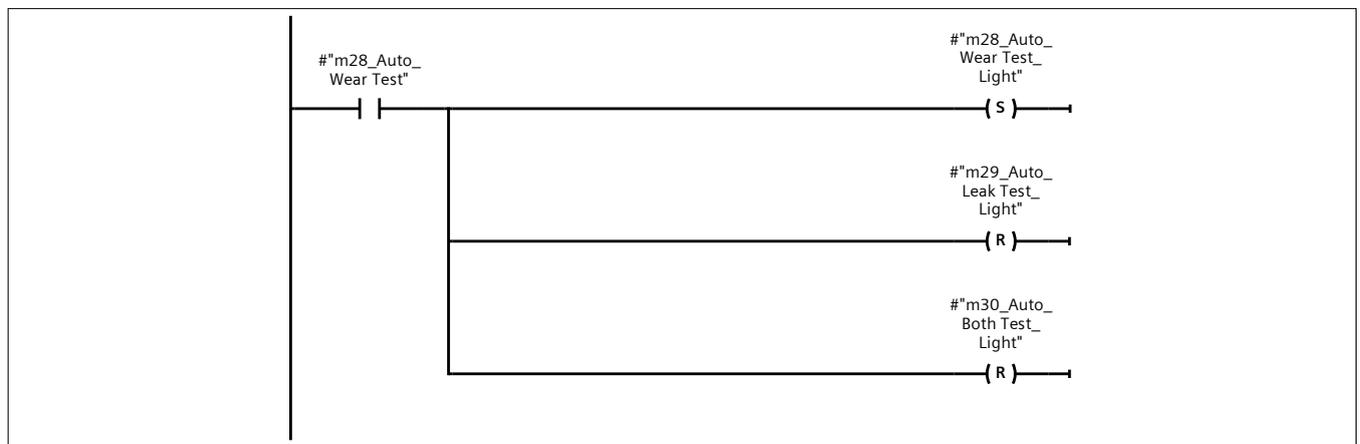
Information

Title	Light around Test Modes Buttons	Author		Comment	
Family		Version	0.1	User-defined ID	

Light around Test Modes Buttons

Name	Data type	Default value	Comment
▼ Input			
m28_Auto_Wear Test	Bool		
m29_Auto_Leak Test	Bool		
m30_Auto_Both Test	Bool		
Reset Cycle Button	Bool		
▼ Output			
m28_Auto_Wear Test_Light	Bool		
m29_Auto_Leak Test_Light	Bool		
m30_Auto_Both Test_Light	Bool		
InOut			
Temp			
Constant			
▼ Return			
Light around Test Modes Buttons	Void		

Network 1: Light set_Wear Mode on HMI



Network 2: Light_Leak Mode on HMI

#"m29_Auto_Leak Test"

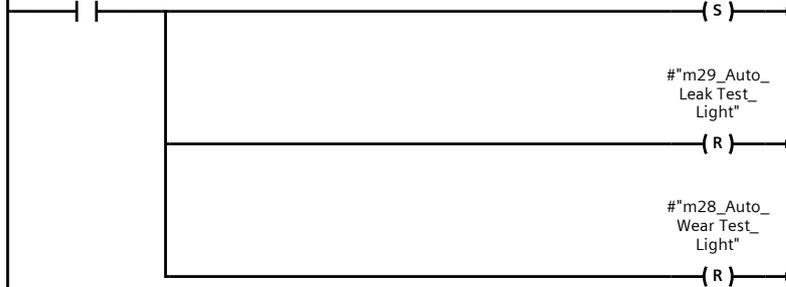
#"m29_Auto_Leak Test_Light"



Network 3: Light_Both Mode on HMI

#"m30_Auto_Both Test"

#"m30_Auto_Both Test_Light"



Network 4: Resrt the Lights in all Modes on HMI

#"Reset Cycle Button"

#"m28_Auto_Wear Test_Light"



The end of the Light Test Mode Block

Automation for All Cyl_22 / PLC_6 [CPU 1512C-1 PN]

PLC tags

Name	Data type	Address	Retain	Access-ible from HMI/O PC UA	Writa-ble from HMI/O PC UA	Visi-ble in HMI engineering	Supervision	Comment
Select leak Test	Bool	%I0.0	False	True	True	True		
Select Wear Test	Bool	%I0.1	False	True	True	True		
Start Button	Bool	%I0.2	False	True	True	True		
Reset Cycle But- ton_Memory	Bool	%M203.1	False	True	True	True		
S0-Cylinder 01	Bool	%I0.4	False	True	True	True		
Door Limit Switch	Bool	%I3.0	False	True	True	True		
Emergency	Bool	%I3.1	False	True	True	True		
Pressure Manometer 01	Int	%IW130	False	True	True	True		
Pressure Manometer 02	Int	%IW132	False	True	True	True		
Pressure Manometer 03	Int	%IW134	False	True	True	True		
Pressure Manometer 04	Int	%IW136	False	True	True	True		
Pressure Manometer 05	Int	%IW138	False	True	True	True		
Pressure Manometer 06	Int	%IW140	False	True	True	True		
Pressure Manometer 07	Int	%IW142	False	True	True	True		
Pressure Manometer 08	Int	%IW144	False	True	True	True		
Pressure Manometer 09	Int	%IW146	False	True	True	True		
Pressure Manometer 10	Int	%IW148	False	True	True	True		
S1-Cylinder 01	Bool	%I0.5	False	True	True	True		
S0-Cylinder 10	Bool	%I2.6	False	True	True	True		
S1-Cylinder 10	Bool	%I2.7	False	True	True	True		
S0-Cylinder 09	Bool	%I2.4	False	True	True	True		
S1-Cylinder 09	Bool	%I2.5	False	True	True	True		
S0-Cylinder 08	Bool	%I2.2	False	True	True	True		
S1-Cylinder 08	Bool	%I2.3	False	True	True	True		
S0-Cylinder 07	Bool	%I2.0	False	True	True	True		
S1-Cylinder 07	Bool	%I2.1	False	True	True	True		
S0-Cylinder 06	Bool	%I1.6	False	True	True	True		
S1-Cylinder 06	Bool	%I1.7	False	True	True	True		
S0-Cylinder 05	Bool	%I1.4	False	True	True	True		
S1-Cylinder 05	Bool	%I1.5	False	True	True	True		
S0-Cylinder 04	Bool	%I1.2	False	True	True	True		

Name	Data type	Address	Retain	Access-ible from HMI/O PC UA	Writa-ble from HMI/O PC UA	Visi-ble in HMI engi-neering	Supervision	Comment
S1-Cylinder 04	Bool	%I1.3	False	True	True	True		
S0-Cylinder 03	Bool	%I1.0	False	True	True	True		
S1-Cylinder 03	Bool	%I1.1	False	True	True	True		
S0-Cylinder 02	Bool	%I0.6	False	True	True	True		
S1-Cylinder 02	Bool	%I0.7	False	True	True	True		
Wear Test LED	Bool	%Q0.0	False	True	True	True		
Leak Test LED	Bool	%Q0.1	False	True	True	True		
Both Test LED	Bool	%Q0.2	False	True	True	True		
Finished Test LED	Bool	%Q0.3	False	True	True	True		
Error LED	Bool	%Q0.4	False	True	True	True		
HPV	Bool	%Q0.5	False	True	True	True		
LPV	Bool	%Q0.6	False	True	True	True		
ABV	Bool	%Q0.7	False	True	True	True		
PBV	Bool	%Q1.0	False	True	True	True		
DV_01	Bool	%Q1.1	False	True	True	True		
DV_02	Bool	%Q1.2	False	True	True	True		
DV_03	Bool	%Q1.3	False	True	True	True		
DV_04	Bool	%Q1.4	False	True	True	True		
DV_05	Bool	%Q1.5	False	True	True	True		
DV_06	Bool	%Q1.6	False	True	True	True		
DV_07	Bool	%Q1.7	False	True	True	True		
DV_08	Bool	%Q2.0	False	True	True	True		
DV_09	Bool	%Q2.1	False	True	True	True		
DV_10	Bool	%Q2.2	False	True	True	True		
Auto_Manu	Bool	%M100.0	False	True	True	True		
Select leak Test_HMI	Bool	%M100.1	False	True	True	True		
Select Wear Test_HMI	Bool	%M100.2	False	True	True	True		
Both Test Mode_HMI	Bool	%M100.3	False	True	True	True		
System_Byte	Byte	%MB1	False	True	True	True		
FirstScan	Bool	%M1.0	False	True	True	True		
DiagStatusUpdate	Bool	%M1.1	False	True	True	True		
AlwaysTRUE	Bool	%M1.2	False	True	True	True		
AlwaysFALSE	Bool	%M1.3	False	True	True	True		
System_Byte(1)	Byte	%MB2	False	True	True	True		
System_Byte(2)	Byte	%MB3	False	True	True	True		
System_Byte(3)	Byte	%MB4	False	True	True	True		
Clock_Byte	Byte	%MB0	False	True	True	True		
Clock_10Hz	Bool	%M0.0	False	True	True	True		
Clock_5Hz	Bool	%M0.1	False	True	True	True		
Clock_2.5Hz	Bool	%M0.2	False	True	True	True		
Clock_2Hz	Bool	%M0.3	False	True	True	True		

Name	Data type	Address	Retain	Access-ible from HMI/O PC UA	Writa-ble from HMI/O PC UA	Visi-ble in HMI engi-neering	Supervision	Comment
Clock_1.25Hz	Bool	%M0.4	False	True	True	True		
Clock_1Hz	Bool	%M0.5	False	True	True	True		
Clock_0.625Hz	Bool	%M0.6	False	True	True	True		
Clock_0.5Hz	Bool	%M0.7	False	True	True	True		
m28_Auto_Wear Test	Bool	%M2.0	False	True	True	True		
m29_Auto_Leak Test	Bool	%M2.1	False	True	True	True		
m30_Auto_Both Test	Bool	%M2.2	False	True	True	True		
m28_Auto_Wear Test_Light	Bool	%M2.3	False	True	True	True		
m29_Auto_Leak Test_Light	Bool	%M2.4	False	True	True	True		
m30_Auto_Both Test_Light	Bool	%M2.5	False	True	True	True		
Leak Test LED_HMI	Bool	%M100.4	False	True	True	True		
Wear Test LED_HMI	Bool	%M100.5	False	True	True	True		
Both Test LED_HMI	Bool	%M100.6	False	True	True	True		
PV_Anterior_Cylinder_01	Real	%MD100	False	True	True	True		
PV_Anterior_Cylinder_03	Real	%MD108	False	True	True	True		
PV_Anterior_Cylinder_04	Real	%MD112	False	True	True	True		
PV_Anterior_Cylinder_05	Real	%MD116	False	True	True	True		
PV_Anterior_Cylinder_06	Real	%MD120	False	True	True	True		
PV_Anterior_Cylinder_07	Real	%MD124	False	True	True	True		
PV_Anterior_Cylinder_08	Real	%MD128	False	True	True	True		
PV_Anterior_Cylinder_09	Real	%MD132	False	True	True	True		
PV_Anterior_Cylinder_10	Real	%MD136	False	True	True	True		
PV_Anterior_Cylinder_02	Real	%MD104	False	True	True	True		
PV_Posterior_Cylinder_01	Real	%MD140	False	True	True	True		
PV_Posterior_Cylinder_02	Real	%MD144	False	True	True	True		
PV_Posterior_Cylinder_03	Real	%MD148	False	True	True	True		
PV_Posterior_Cylinder_04	Real	%MD152	False	True	True	True		
PV_Posterior_Cylinder_05	Real	%MD156	False	True	True	True		
PV_Posterior_Cylinder_06	Real	%MD160	False	True	True	True		
PV_Posterior_Cylinder_07	Real	%MD164	False	True	True	True		

Name	Data type	Address	Retain	Access-ible from HMI/O PC UA	Writa-ble from HMI/O PC UA	Visi-ble in HMI engi-neering	Supervision	Comment
 PV_Posterior_Cylinder_08	Real	%MD168	False	True	True	True		
 PV_Posterior_Cylinder_09	Real	%MD172	False	True	True	True		
 PV_Posterior_Cylinder_10	Real	%MD176	False	True	True	True		
 ON-OFF_HMI Cylinder 1	Bool	%M101.0	False	True	True	True		
 ON-OFF_HMI Cylinder 2	Bool	%M101.1	False	True	True	True		
 ON-OFF_HMI Cylinder 3	Bool	%M101.2	False	True	True	True		
 ON-OFF_HMI Cylinder 4	Bool	%M101.3	False	True	True	True		
 ON-OFF_HMI Cylinder 5	Bool	%M101.4	False	True	True	True		
 ON-OFF_HMI Cylinder 6	Bool	%M101.5	False	True	True	True		
 ON-OFF_HMI Cylinder 7	Bool	%M101.6	False	True	True	True		
 ON-OFF_HMI Cylinder 8	Bool	%M101.7	False	True	True	True		
 ON-OFF_HMI Cylinder 9	Bool	%M102.0	False	True	True	True		
 ON-OFF_HMI Cylinder 10	Bool	%M102.1	False	True	True	True		
 Screen_Button & Indicators	Bool	%M102.2	False	True	True	True		
 Ended Wear Cycles_Cylinder_1	Bool	%M5.0	False	True	True	True		
 Ended Wear Cycles_Cylinder_2	Bool	%M5.1	False	True	True	True		
 Ended Wear Cycles_Cylinder_3	Bool	%M5.2	False	True	True	True		
 Ended Wear Cycles_Cylinder_4	Bool	%M5.3	False	True	True	True		
 Ended Wear Cycles_Cylinder_5	Bool	%M5.4	False	True	True	True		
 Ended Wear Cycles_Cylinder_6	Bool	%M5.5	False	True	True	True		
 Ended Wear Cycles_Cylinder_7	Bool	%M5.6	False	True	True	True		
 Ended Wear Cycles_Cylinder_8	Bool	%M5.7	False	True	True	True		
 Ended Wear Cycles_Cylinder_9	Bool	%M6.0	False	True	True	True		
 Ended Wear Cycles_Cylinder_10	Bool	%M6.1	False	True	True	True		
 Priority_Wear & Leak	Bool	%M200.0	False	True	True	True		
 Reset Count CYCLE_HMI_All Cylinders	Bool	%M200.2	False	True	True	True		

Name	Data type	Address	Retain	Acces- sible from HMI/O PC UA	Writa- ble from HMI/O PC UA	Visi- ble in HMI engi- neering	Supervision	Comment
Reset Cycle Total_HMI	Bool	%M200.3	False	True	True	True		
Reset Cycle_Cylinder 1	Bool	%M200.4	False	True	True	True		
Reset Cycle_Cylinder 2	Bool	%M200.5	False	True	True	True		
Reset Cycle_Cylinder 3	Bool	%M200.6	False	True	True	True		
Reset Cycle_Cylinder 4	Bool	%M200.7	False	True	True	True		
Reset Cycle_Cylinder 5	Bool	%M201.0	False	True	True	True		
Reset Cycle_Cylinder 6	Bool	%M201.1	False	True	True	True		
Reset Cycle_Cylinder 7	Bool	%M201.2	False	True	True	True		
Reset Cycle_Cylinder 8	Bool	%M201.3	False	True	True	True		
Reset Cycle_Cylinder 9	Bool	%M201.4	False	True	True	True		
Reset Cycle_Cylinder 10	Bool	%M201.5	False	True	True	True		
Reset Count_Cylinder 1	Bool	%M201.6	False	True	True	True		
Reset Count_Cylinder 2	Bool	%M201.7	False	True	True	True		
Reset Count_Cylinder 3	Bool	%M202.0	False	True	True	True		
Reset Count_Cylinder 4	Bool	%M202.1	False	True	True	True		
Reset Count_Cylinder 5	Bool	%M202.2	False	True	True	True		
Reset Count_Cylinder 6	Bool	%M202.3	False	True	True	True		
Reset Count_Cylinder 7	Bool	%M202.4	False	True	True	True		
Reset Count_Cylinder 8	Bool	%M202.5	False	True	True	True		
Reset Count_Cylinder 9	Bool	%M202.6	False	True	True	True		
Reset Count_Cylinder 10	Bool	%M202.7	False	True	True	True		
Reset Count Master_HMI	Bool	%M203.0	False	True	True	True		
Reset Cycle Button	Bool	%I0.3	False	True	True	True		
Start Button_HMI	Bool	%M203.2	False	True	True	True		

The end of the PLC tag table