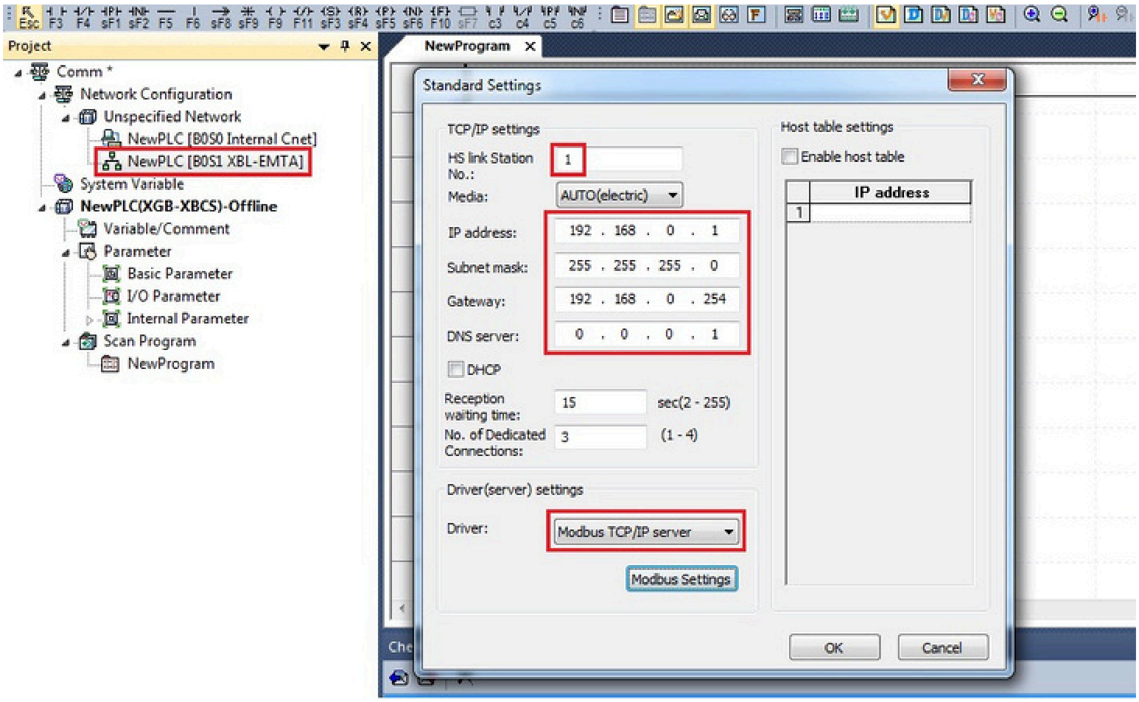
INDEX

APPLICATIONS

Ethernet TCP/IP COMMUNICATION BETWEEN 4 PLC'S:

01- PLC 01 SLAVE CONFIGURATION:

1. With the project created, recognize the modules:
2. Online > Connect;
3. Set the PLC to STOP;
   Online > Diagnosis > I/O Information...;
4. Click on "I/O Sync" and confirm with "OK";
5. Double-click on NewPLC [B0S1 XBL-EMTA];
6. The following screen will open:



Standard Settings

TCP/IP settings

HS link Station No.: 1

Media: AUTO(electric)

IP address: 192 . 168 . 0 . 1

Subnet mask: 255 . 255 . 255 . 0

Gateway: 192 . 168 . 0 . 254

DNS server: 0 . 0 . 0 . 1

□ DHCP

Reception waiting time: 15 sec(2 - 255)

No. of Dedicated Connections: 3 (1 - 4)

Driver(server) settings

Driver: Modbus TCP/IP server

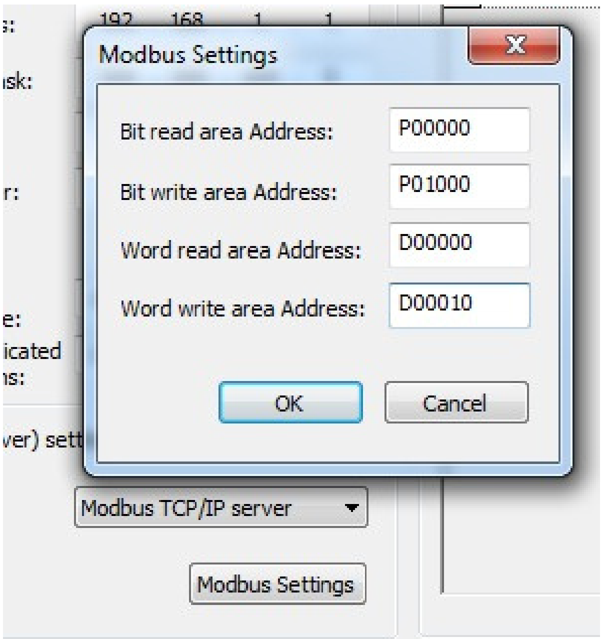
Host table settings

□ Enable host table

	IP address
1	

Configure the parameters as shown in the screen above.

7. Click on Modbus Settings:



Modbus Settings

Bit read area Address: P00000

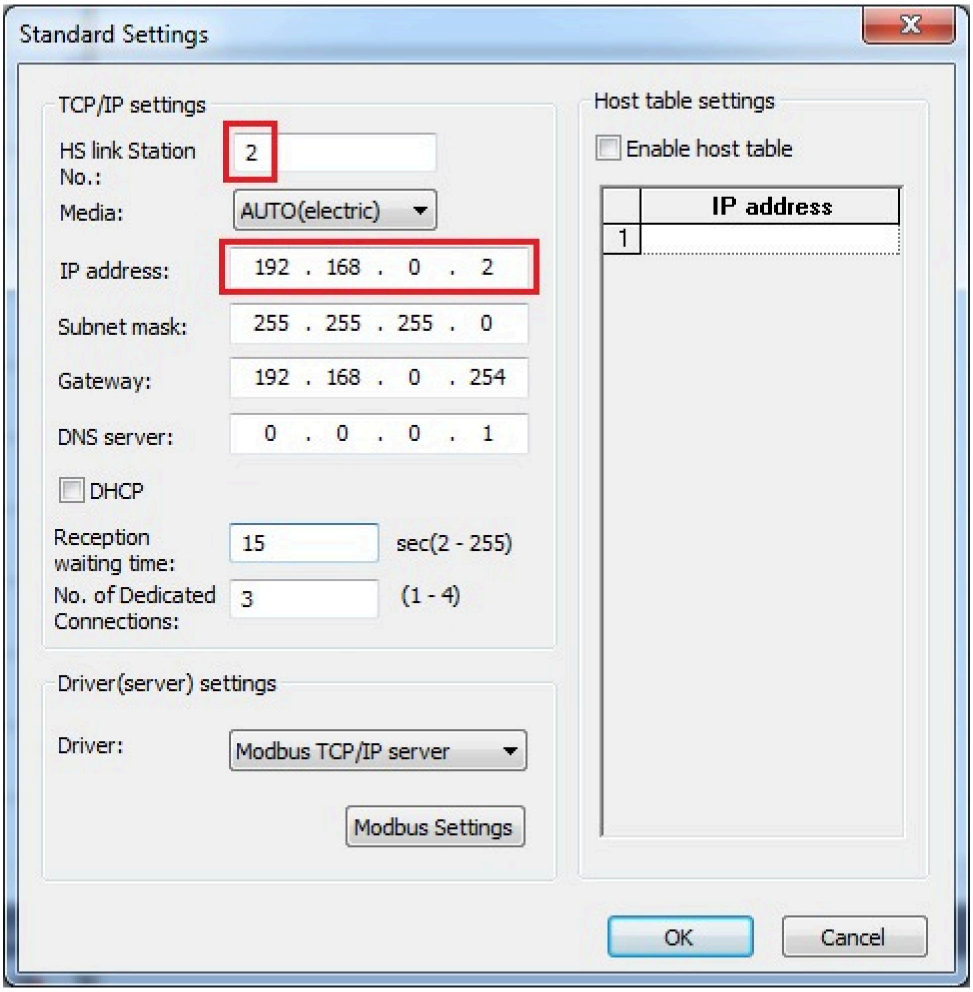
Bit write area Address: P01000

Word read area Address: D00000

Word write area Address: D00010

Configure the parameters as shown in the screen above. This means that the information made available by the Slave PLC for reading must be placed in the D0 memory of the Slave PLC. When the Master PLC requests reading, the data in the D0 memory of the Slave PLC will be transferred to a memory in the Master PLC. When the Master PLC writes some data to the Slave PLC, this data will appear in the D10 memory of the Slave PLC.

8. **For the other PLCs, the previous steps must be repeated, changing only the Station Number and IP address for each PLC, as shown in the screen below:**



Standard Settings

TCP/IP settings

HS link Station No.: 2

Media: AUTO(electric)

IP address: 192 . 168 . 0 . 2

Subnet mask: 255 . 255 . 255 . 0

Gateway: 192 . 168 . 0 . 254

DNS server: 0 . 0 . 0 . 1

□ DHCP

Reception waiting time: 15 sec(2 - 255)

No. of Dedicated Connections: 3 (1 - 4)

Driver(server) settings

Driver: Modbus TCP/IP server

Host table settings

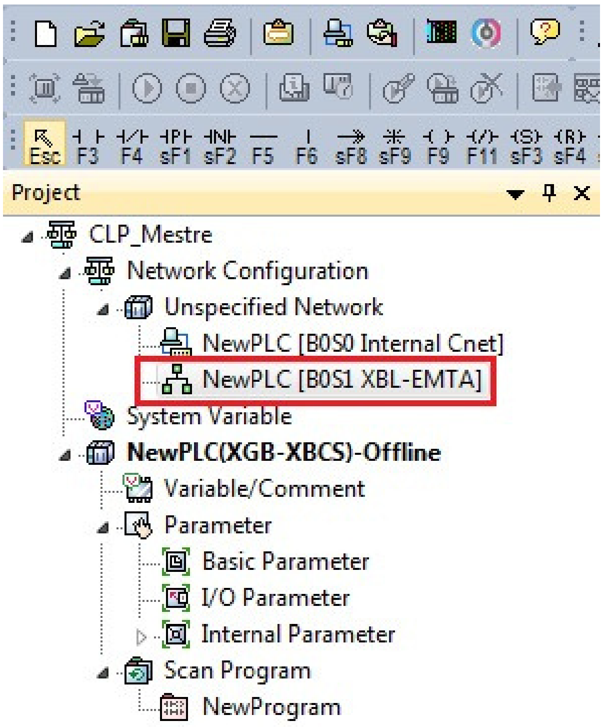
□ Enable host table

	IP address
1	

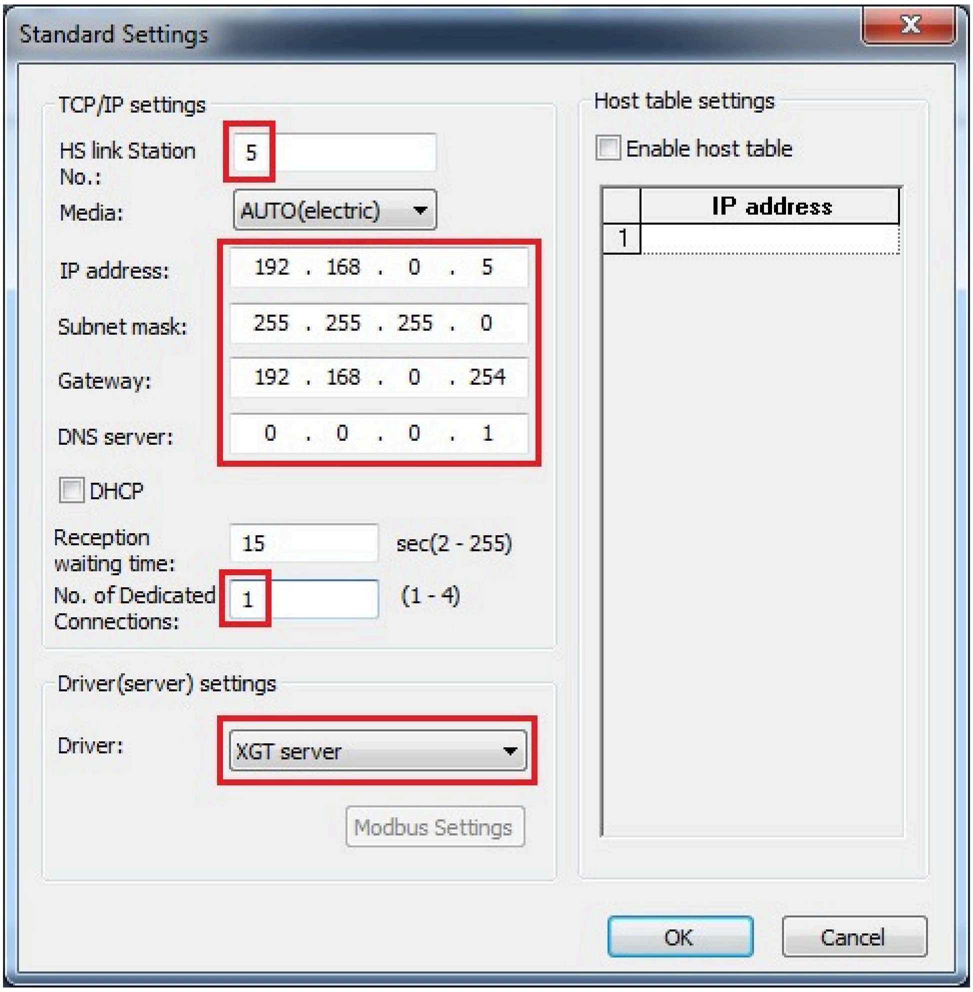
9. Click on Online > Write…;

02- MASTER PLC CONFIGURATION:

1. With the project created, recognize the modules: Online >
2. Connect; Put the PLC in STOP; Online > Diagnosis > I/O
3. Information...; Click on “I/O Sync” and confirm with “OK”; Double
4. click on NewPLC [B0S1 XBL-EMTA];
5.



1- **Configure the data as shown in the screen below:**



Standard Settings

TCP/IP settings

HS link Station No.: 5

Media: AUTO(electric)

IP address: 192 . 168 . 0 . 5

Subnet mask: 255 . 255 . 255 . 0

Gateway: 192 . 168 . 0 . 254

DNS server: 0 . 0 . 0 . 1

□ DHCP

Reception waiting time: 15 sec(2 - 255)

No. of Dedicated Connections: 1 (1 - 4)

Driver(server) settings

Driver: XGT server

Host table settings

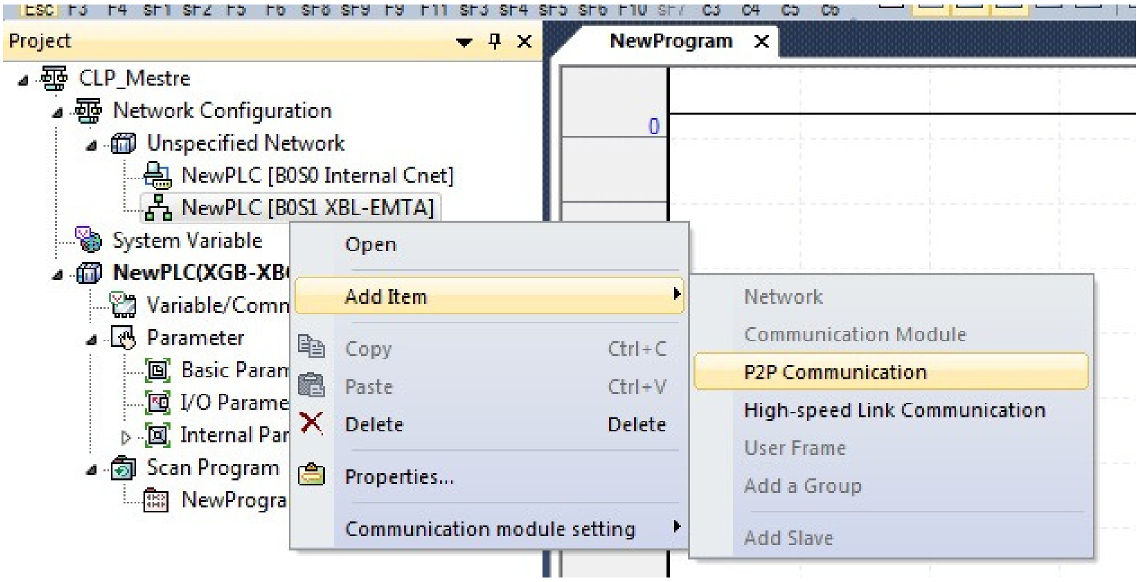
□ Enable host table

	IP address
1	

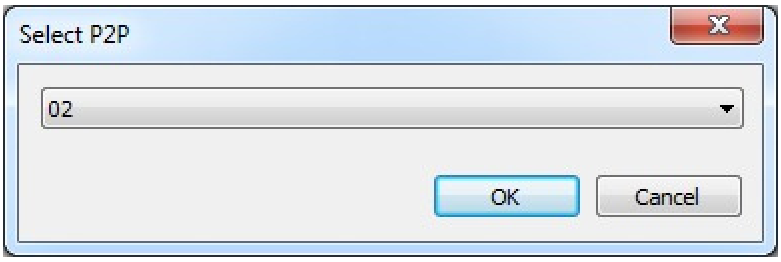
Remember that the IP range of all PLCs must be the same. In this example, our Master PLC ended in 5.

The “No of Dedicated Connection” needs to be set to 1 for 3 slave communication.

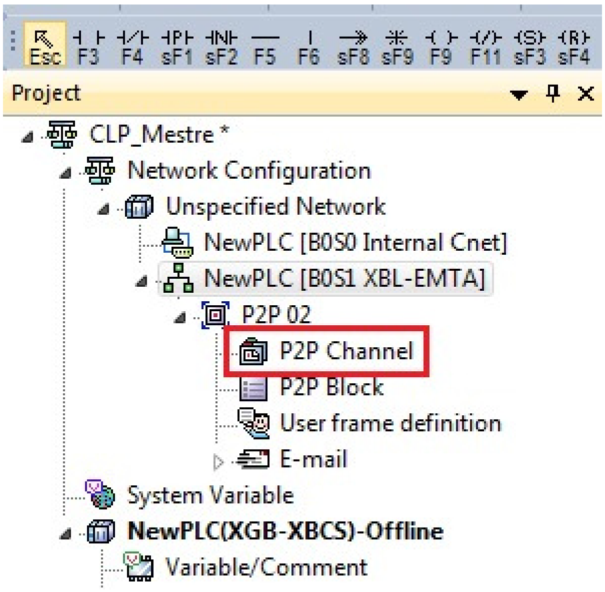
2- **Right-click NewPLC [B0S1 XBL-EMTA] > Add Item > P2P Communication:**



3- **Select the Slot and confirm with “OK”;**

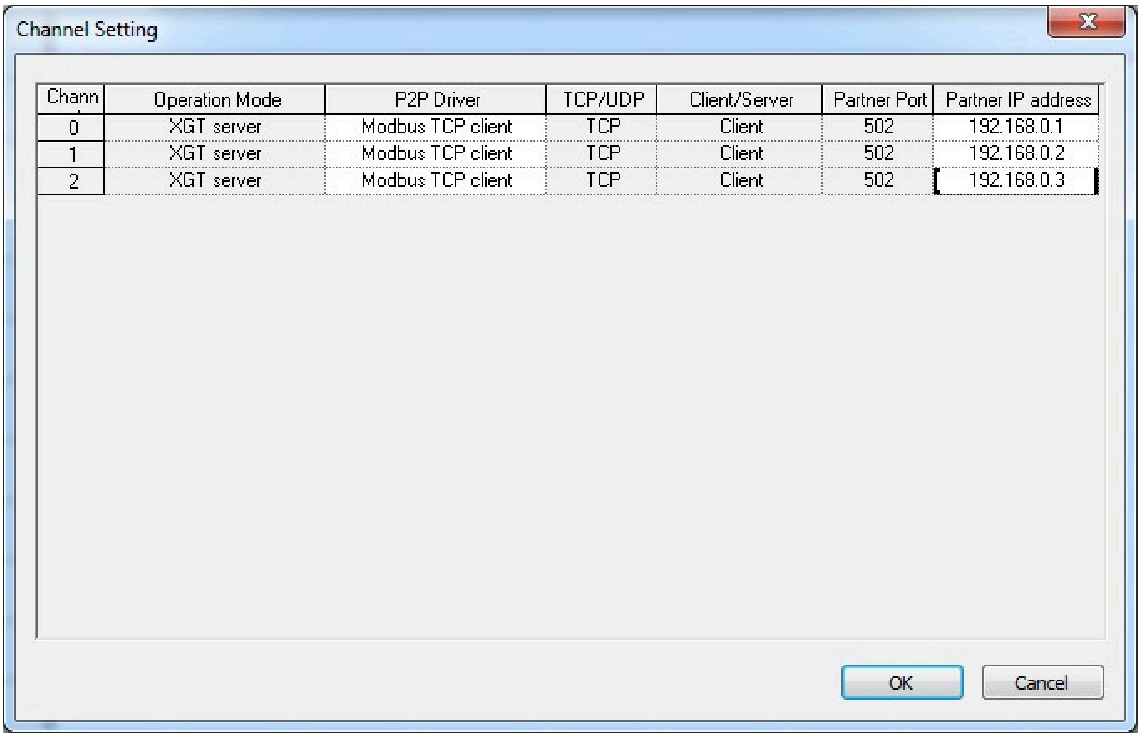


4- Click 2 vezes em P2P Channel:



- CLP_Mestre *
  - Network Configuration
    - Unspecified Network
      - NewPLC [B0S0 Internal Cnet]
      - NewPLC [B0S1 XBL-EMTA]
        - P2P 02
          - P2P Channel
          - P2P Block
          - User frame definition
          - E-mail
  - System Variable
  - **NewPLC(XGB-XBCS)-Offline**
    - Variable/Comment

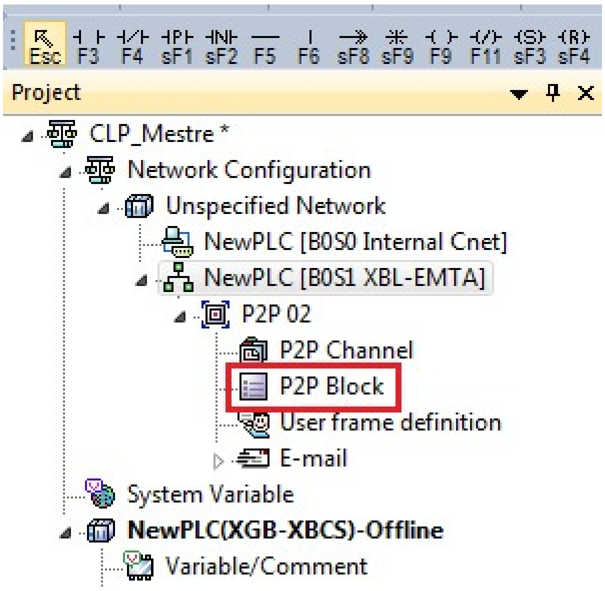
5- Configure as shown in the screen below, placing the IP addresses previously configured on the Slaves:



Channel Setting

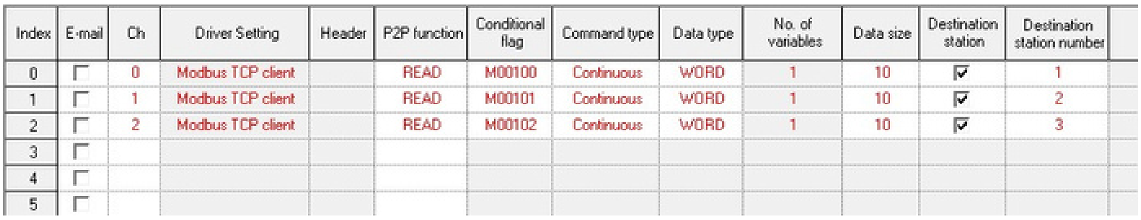
Chann	Operation Mode	P2P Driver	TCP/UDP	Client/Server	Partner Port	Partner IP address
0	XGT server	Modbus TCP client	TCP	Client	502	192.168.0.1
1	XGT server	Modbus TCP client	TCP	Client	502	192.168.0.2
2	XGT server	Modbus TCP client	TCP	Client	502	192.168.0.3

6- **Click 2 vezes em P2P Block:**



- CLP_Mestre *
  - Network Configuration
    - Unspecified Network
      - NewPLC [B0S0 Internal Cnet]
      - NewPLC [B0S1 XBL-EMTA]
        - P2P 02
          - P2P Channel
          - P2P Block
          - User frame definition
          - E-mail
  - System Variable
  - **NewPLC(XGB-XBCS)-Offline**
    - Variable/Comment

7- Now we will configure all the READ addresses of the Master PLC:



Index	E-mail	Ch	Driver Setting	Header	P2P function	Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number
0	☐	0	Modbus TCP client		READ	M00100	Continuous	WORD	1	10	☑	1
1	☐	1	Modbus TCP client		READ	M00101	Continuous	WORD	1	10	☑	2
2	☐	2	Modbus TCP client		READ	M00102	Continuous	WORD	1	10	☑	3
3	☐											
4	☐											
5	☐											

Ch / Driver Settings – *Configured in step 5;*

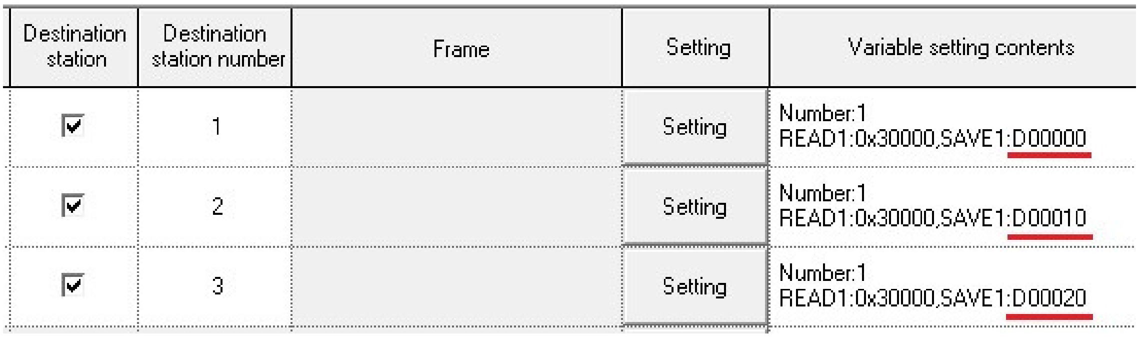
P2P – *Configured for data reading;*

Conditional Flag – *Every time memories M100, M101, M102 go to high logic level the Master PLC performs the reading;*

Data Size – *Space reserved for data, i.e., 10 words;*

Destination Station Number – *Configured in step 8 of item 01 - PLC 01 Slave Configuration;*

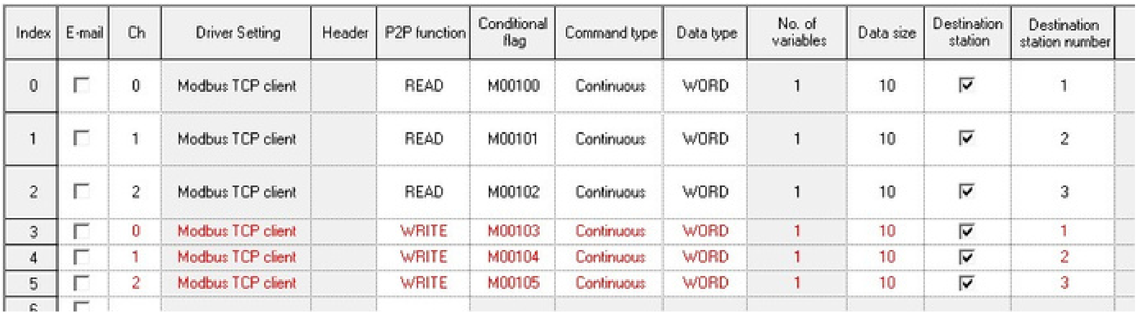
8- Click on settings and configure the reading memories. In this example, we configured the memories: D0, D10 and D20. This means that these memories are the ones that will receive data stored in the D0 memory of the Slave PLCs.



Destination station	Destination station number	Frame	Setting	Variable setting contents
☑	1		Setting	Number:1 READ1:0x30000,SAVE1:D00000
☑	2		Setting	Number:1 READ1:0x30000,SAVE1:D00010
☑	3		Setting	Number:1 READ1:0x30000,SAVE1:D00020

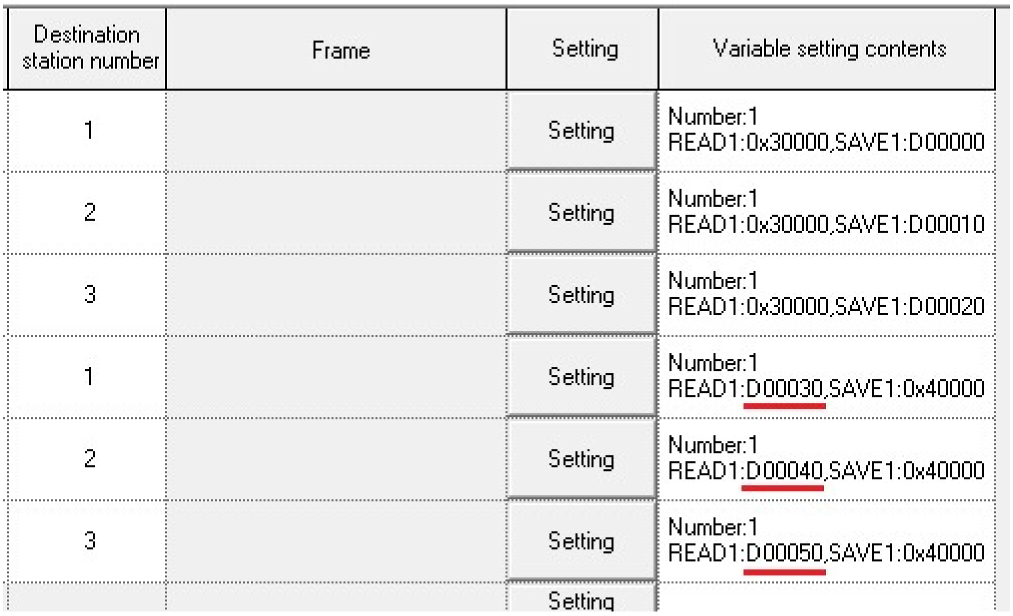
- D0 to D9 – Receiving Data from Slave 1’s D0 Memory; - D10 to D19 – Receiving Data from Slave 2’s D0 Memory; - D20 to D29 – Receiving Data from Slave 3’s D0 Memory;

9- Now we will configure all the WRITE addresses of the Master PLC:



Index	E-mail	Ch	Driver Setting	Header	P2P function	Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number
0	□	0	Modbus TCP client		READ	M00100	Continuous	WORD	1	10	☑	1
1	□	1	Modbus TCP client		READ	M00101	Continuous	WORD	1	10	☑	2
2	□	2	Modbus TCP client		READ	M00102	Continuous	WORD	1	10	☑	3
3	□	0	Modbus TCP client		WRITE	M00103	Continuous	WORD	1	10	☑	1
4	□	1	Modbus TCP client		WRITE	M00104	Continuous	WORD	1	10	☑	2
5	□	2	Modbus TCP client		WRITE	M00105	Continuous	WORD	1	10	☑	3
6	□											

10- Click on settings and configure the writing memories. In this example we configured the memories: D30, D40 and D50. This means that these memories are the memories that will send the data to the D10 memory of the Slave PLCs.



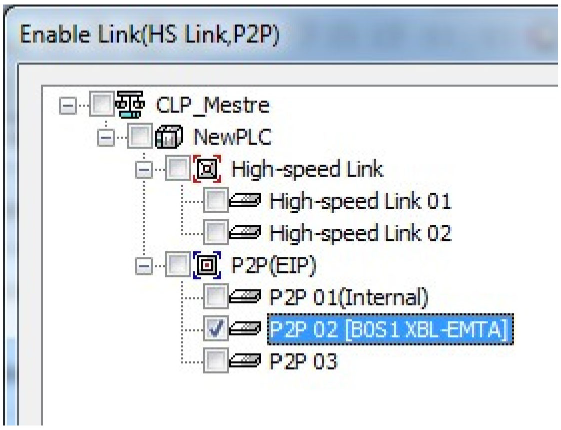
Destination station number	Frame	Setting	Variable setting contents
1		Setting	Number:1 READ1:0x30000,SAVE1:D00000
2		Setting	Number:1 READ1:0x30000,SAVE1:D00010
3		Setting	Number:1 READ1:0x30000,SAVE1:D00020
1		Setting	Number:1 READ1:D00030,SAVE1:0x40000
2		Setting	Number:1 READ1:D00040,SAVE1:0x40000
3		Setting	Number:1 READ1:D00050,SAVE1:0x40000
		Setting	

- D30– Sending Data to Memory D10 of Slave 1; - D40– Sending Data to Memory D10 of Slave 2; - D50– Sending Data to Memory D10 of Slave 3;

11- Click em Online > Write;

12- Click em Online > Communication Module Setting > Enable Link (HS Link, P2P);

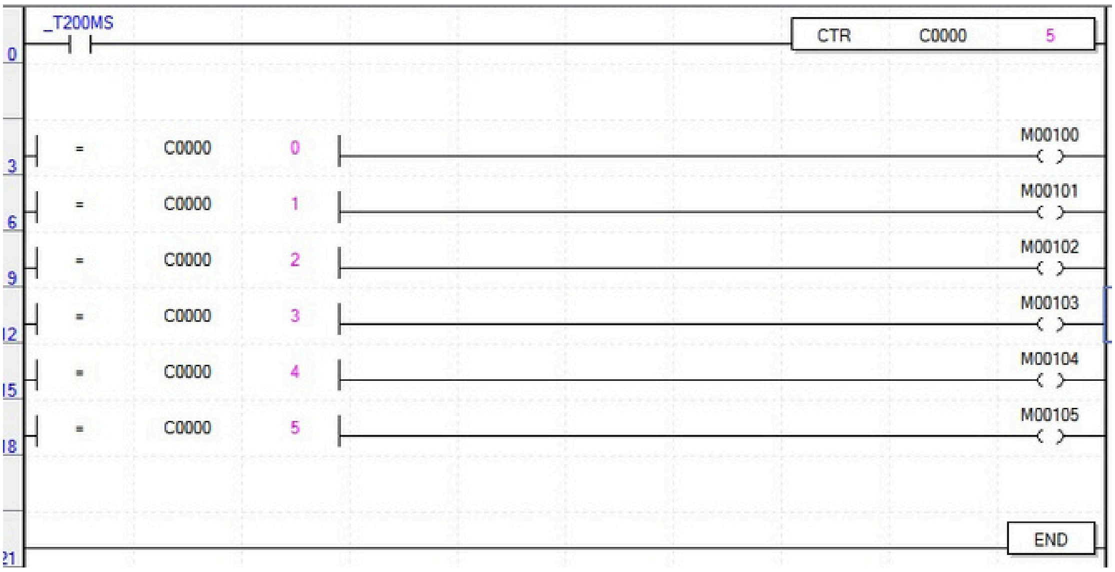
13- Select the option P2P 02 [B0S1 XBL-EMTA] e click em Write > Ok > Close:



Enable Link(HS Link,P2P)

- □ CLP_Mestre
  - □ NewPLC
    - □ High-speed Link
      - □ High-speed Link 01
      - □ High-speed Link 02
    - □ P2P(EIP)
      - □ P2P 01(Internal)
      - ☑ P2P 02 [B0S1 XBL-EMTA]
      - □ P2P 03

14- automatic recall of “Condition Flags” memories: automatic recall of “Condition Flags” memories:



Transfer this program to the Master PLC;

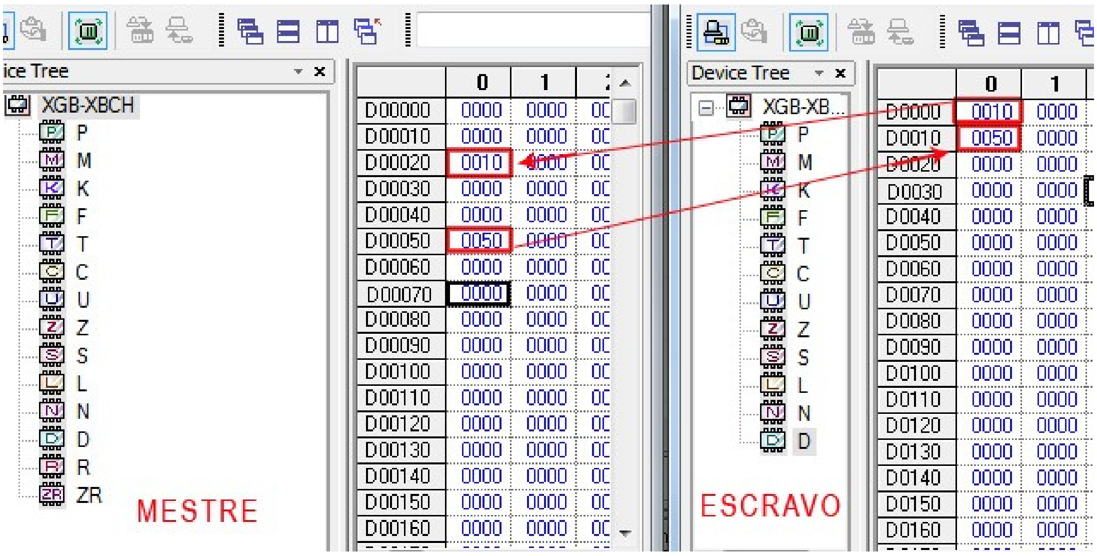
15- 15- To perform a test, open two XG5000, one with the Master's program and the other with the program of one of the Slaves;

16. In the Master PLC program, click on Monitor > Device Monitoring and select D;

17. In the Slave PLC program, click on Monitor > Device Monitoring and select D;

18. In the Master PLC's Device Monitoring, enter a value in memory D30, D40, or D50. This value will be read by the Slave and will appear in memory D10 of the respective slave.

19- In one of the slaves, enter a value in memory D0 and this value will be read by the Master and will appear in one of the memories D0, D10 or D20 depending on the slave.



	0	1	2
D00000	0000	0000	0C
D00010	0000	0000	0C
D00020	0010	0000	0C
D00030	0000	0000	0C
D00040	0000	0000	0C
D00050	0050	0000	0C
D00060	0000	0000	0C
D00070	0000	0000	0C
D00080	0000	0000	0C
D00090	0000	0000	0C
D00100	0000	0000	0C
D00110	0000	0000	0C
D00120	0000	0000	0C
D00130	0000	0000	0C
D00140	0000	0000	0C
D00150	0000	0000	0C
D00160	0000	0000	0C

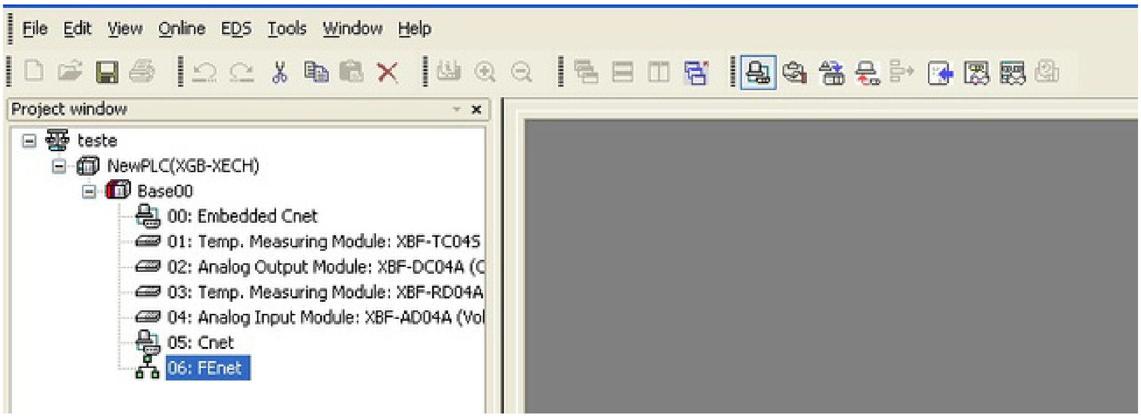
	0	1
D0000	0010	0000
D0010	0050	0000
D0020	0000	0000
D0030	0000	0000
D0040	0000	0000
D0050	0000	0000
D0060	0000	0000
D0070	0000	0000
D0080	0000	0000
D0090	0000	0000
D0100	0000	0000
D0110	0000	0000
D0120	0000	0000
D0130	0000	0000
D0140	0000	0000
D0150	0000	0000
D0160	0000	0000

In the example above, the value 50 was written to the D50 memory of the Master PLC and this value was read (transferred) by the D10 memory of Slave 3. The value 10 was also written to the D0 memory of Slave 3 and this value was read (transferred) by the D50 memory of the Master.

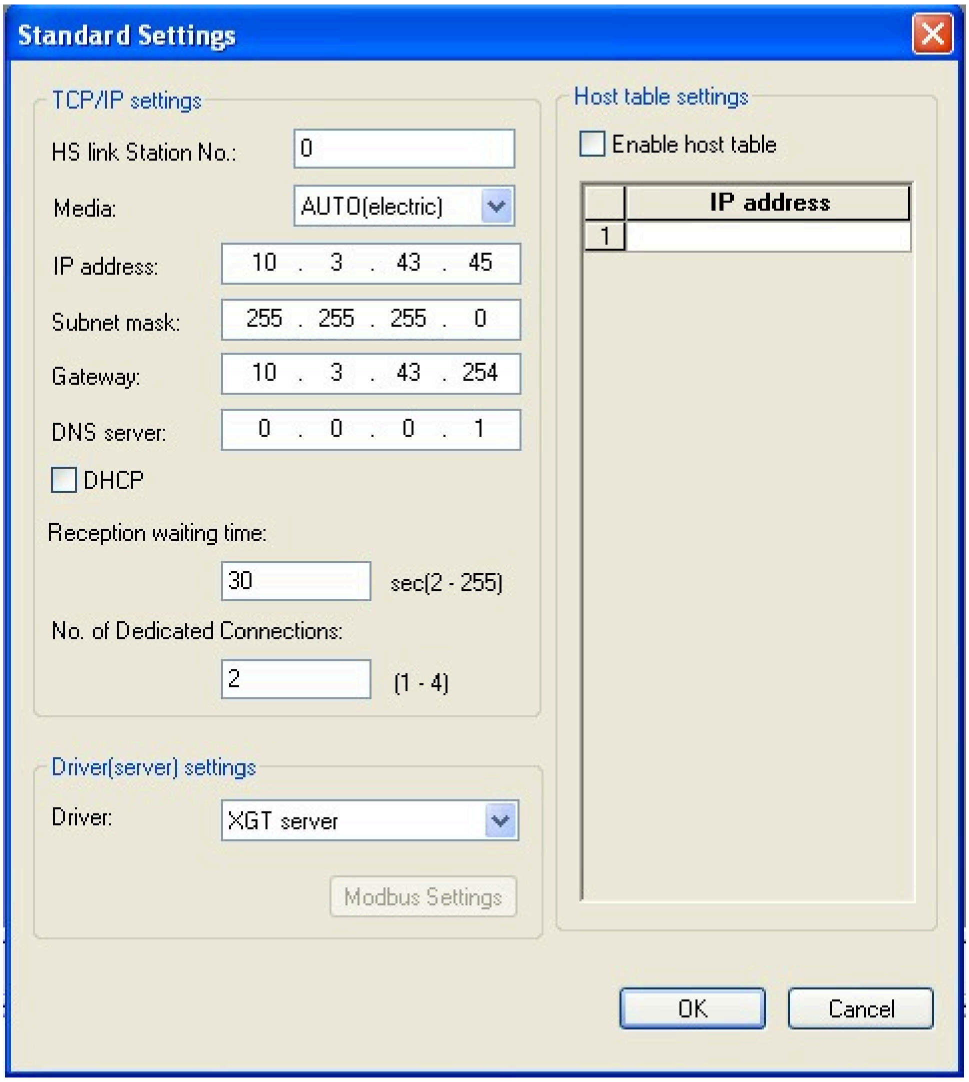
END.

Ethernet COMMUNICATION: PLC XGB / XBL EMTA

- On the XG5000: 1. Click on Tools > Network Manager to open XG-PD; 2. New > Choose a name > Choose the PLC model; 3. Click on Online > Conect; 4. Click on Online > Read IO Information; 5. Double-click on FEnet;


File
Edit
View
Online
EDS
Tools
Window
Help
Project window
teste
NewPLC(XGB-XECH)
Base00
00: Embedded Cnet
01: Temp. Measuring Module: XBF-TC04S
02: Analog Output Module: XBF-DC04A (C
03: Temp. Measuring Module: XBF-RD04A
04: Analog Input Module: XBF-AD04A (Vol
05: Cnet
06: FEnet

6. **Configure the parameters that will be used for the XBL-EMTA module, as shown in the screen below and click OK:**



Standard Settings

TCP/IP settings

HS link Station No.: 0

Media: AUTO(electric)

IP address: 10 . 3 . 43 . 45

Subnet mask: 255 . 255 . 255 . 0

Gateway: 10 . 3 . 43 . 254

DNS server: 0 . 0 . 0 . 1

□ DHCP

Reception waiting time: 30 sec(2 - 255)

No. of Dedicated Connections: 2 (1 - 4)

Driver(server) settings

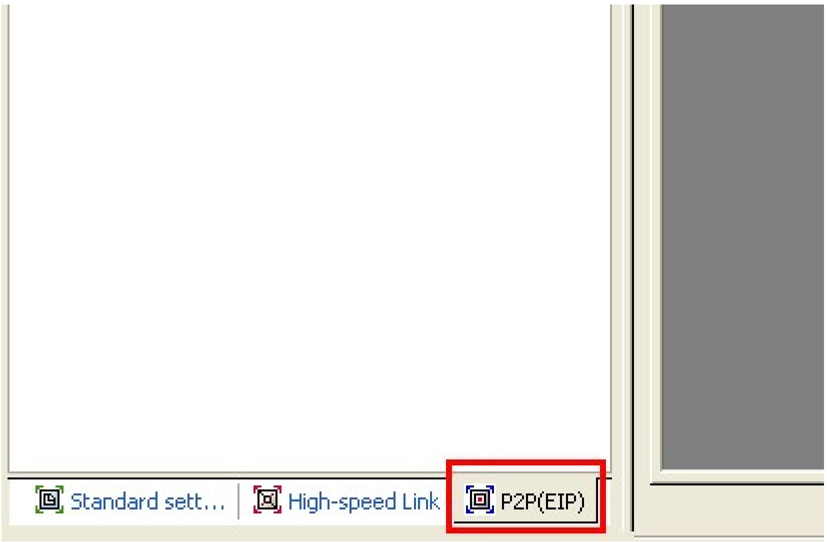
Driver: XGT server

Host table settings

□ Enable host table

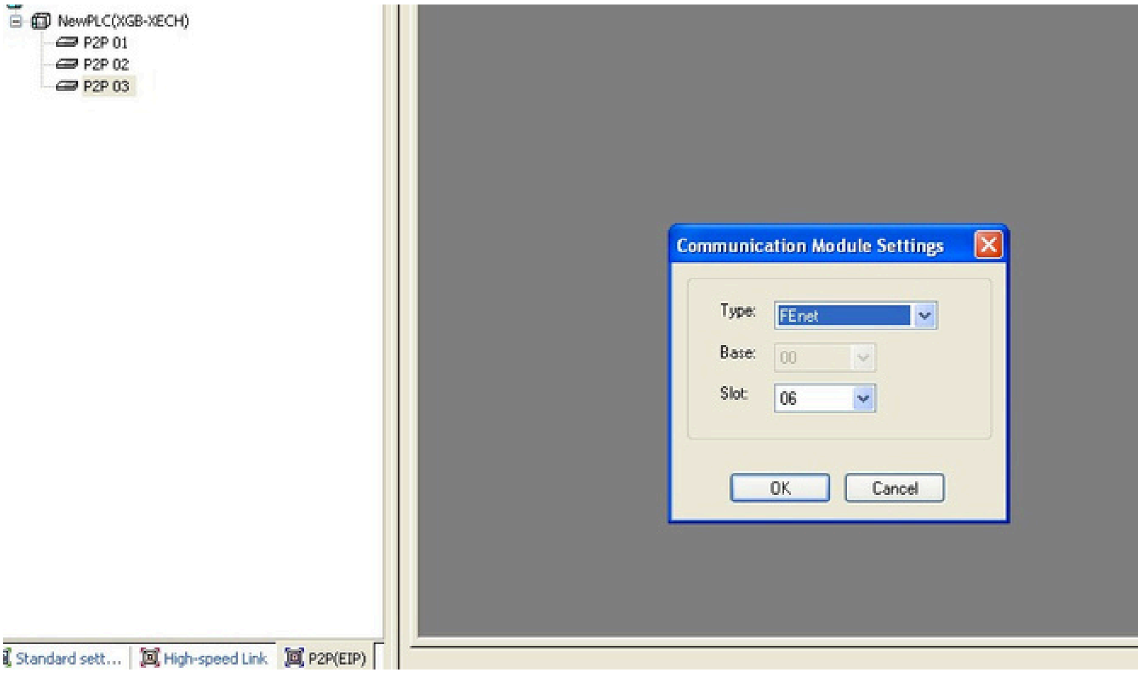
	IP address
1	

7- Click on P2P(EIP) at the bottom left of the screen;



8- Double-click on P2P 03 and select FEnet and the slot where the module is located, in this case

example, slot 6. Click OK:



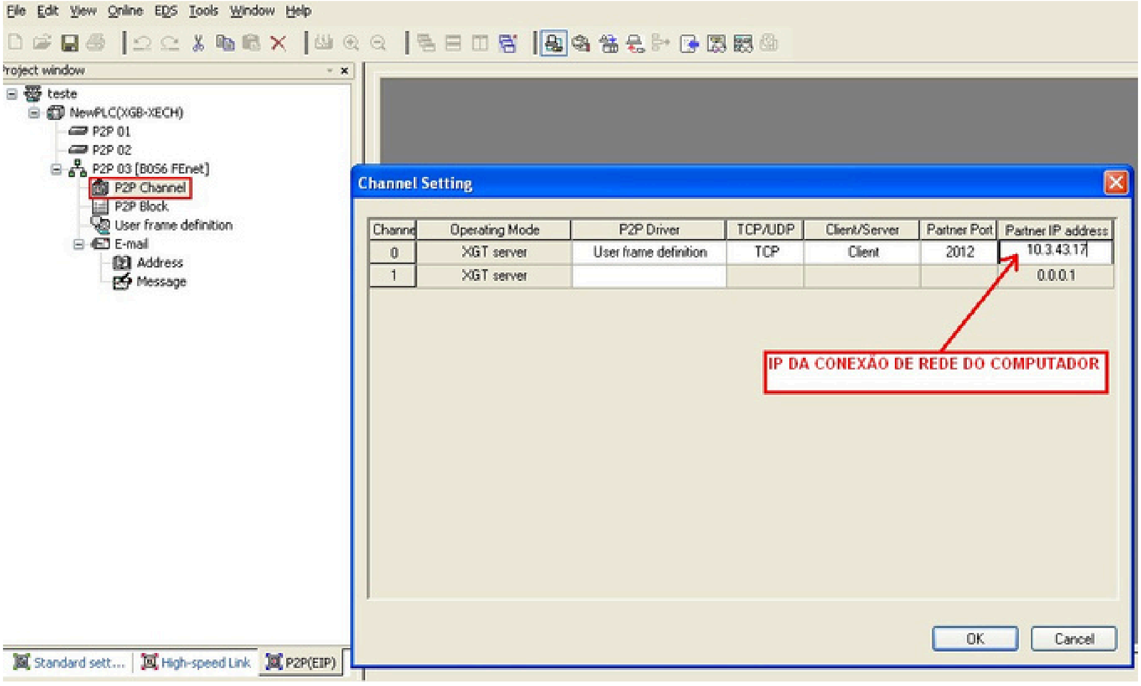
Communication Module Settings

Type: FEnet

Base: 00

Slot: 06

9- Double click on P2P Chanel and configure the parameters as shown in the screen below:

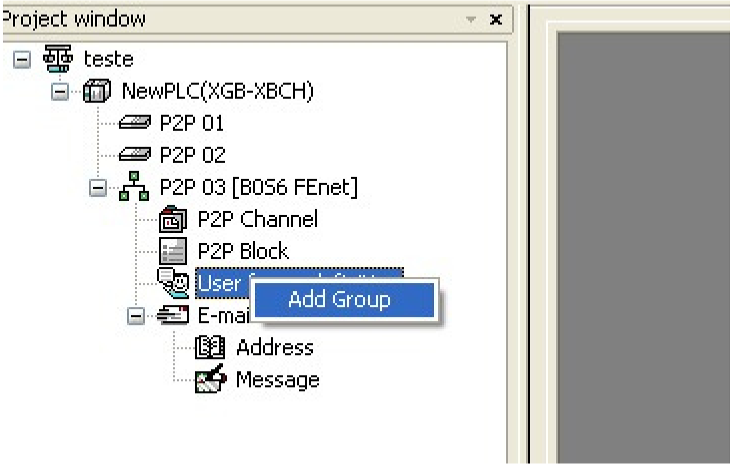


Channel Setting

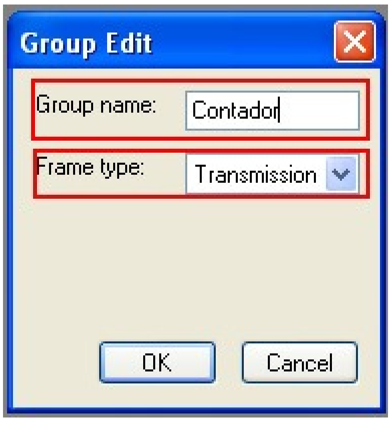
Channe	Operating Mode	P2P Driver	TCP/UDP	Client/Server	Partner Port	Partner IP address
0	XGT server	User frame definition	TCP	Client	2012	10.3.43.17
1	XGT server					0.0.0.1

IP DA CONEXÃO DE REDE DO COMPUTADOR

10 – Right click on User frame definition > Add Group:



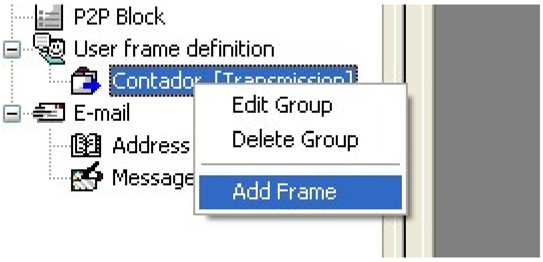
11- Preencha o Group name e selecione em Frame type: Transmission, pois nesse caso queremos enviar dados do CLP para o Computador via Ethernet:



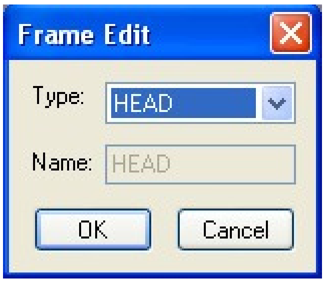
Group name: Contador

Frame type: Transmission

12 – Right-click on [Transmission] Counter and select Add Frame:



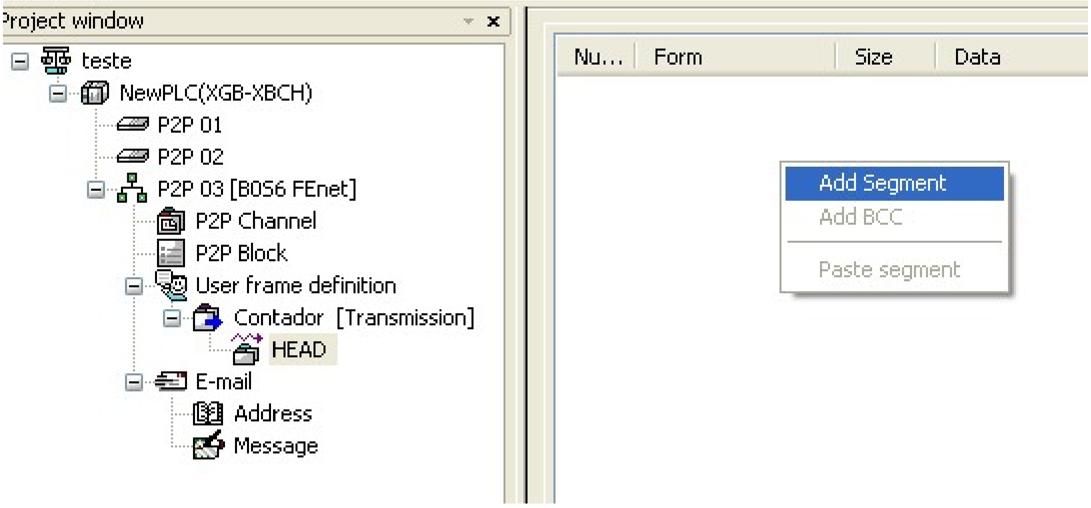
13 – Set the parameters as shown in the screen below and click OK:



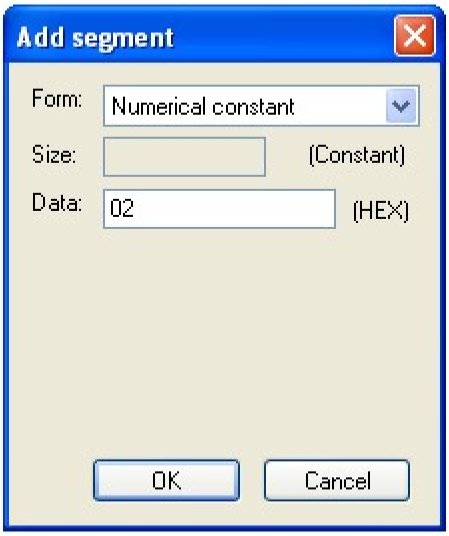
Type: HEAD

Name: HEAD

14- Right click on the white screen on the right and select Add Segment:



15- Configure the Parameters as shown in the screen below:



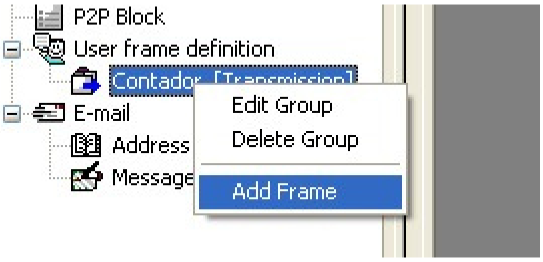
Form: Numerical constant

Size: (Constant)

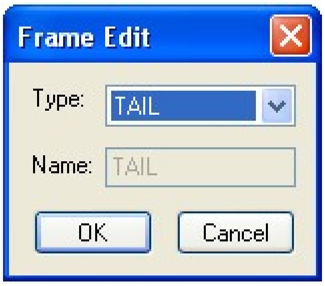
Data: 02 (HEX)

02: STX in Hexadecimal;

16 – Right-click on Counter [Transmission] again and select Add Frame:



17 – Set the parameters as shown in the screen below and click OK:

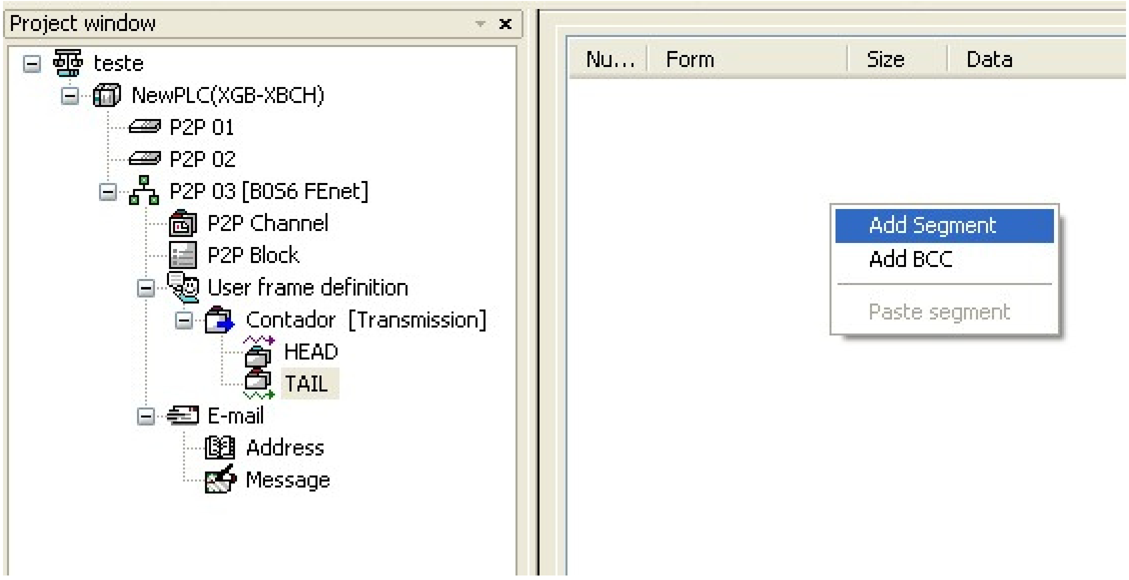


Frame Edit

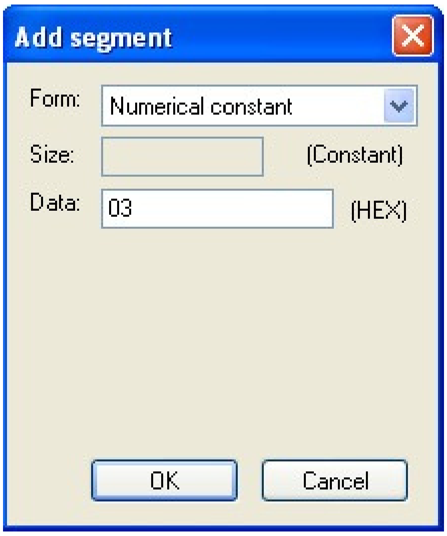
Type: TAIL

Name: TAIL

18- Right click on the white screen on the right and select Add Segment:



19- Configure os Parâmetros conforme a tela abaixo:



Add segment

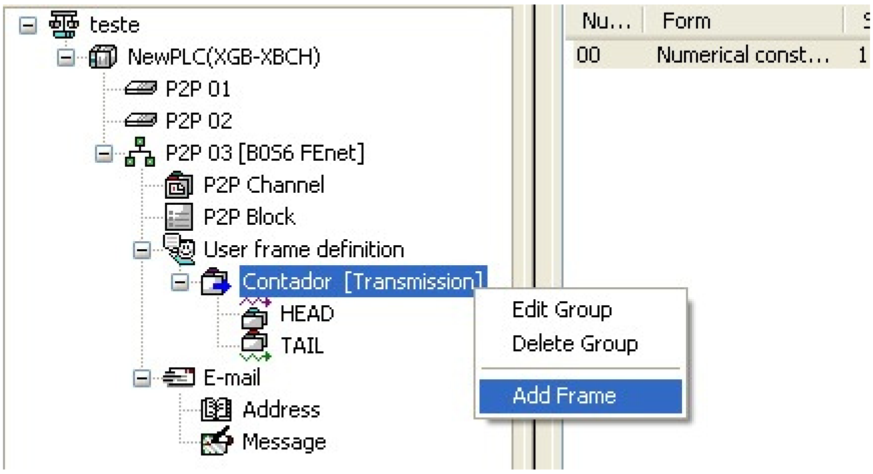
Form: Numerical constant

Size: (Constant)

Data: 03 (HEX)

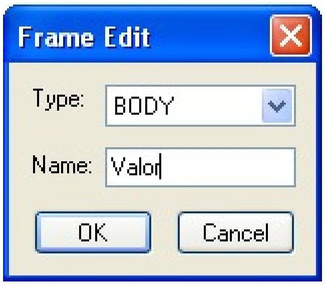
03: ETX em Hexadecimal;

20 – Click novamente com o botão direito em Contador [Transmission] selecione Add Frame:



Nu...	Form	S
00	Numerical const...	1

21 – Set the parameters as shown in the screen below and click OK

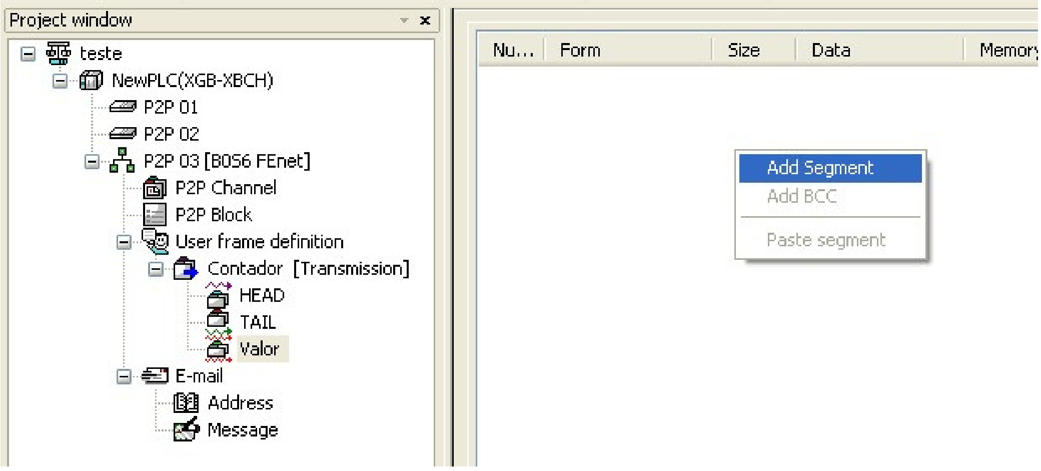


Frame Edit

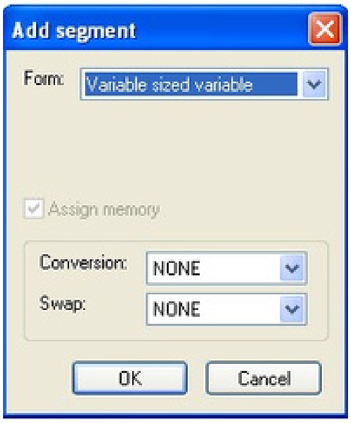
Type: BODY

Name: Valor

22- Right click on the white screen on the right and select Add Segment:



23- Configure the Parameters as shown in the screen below:



Add segment

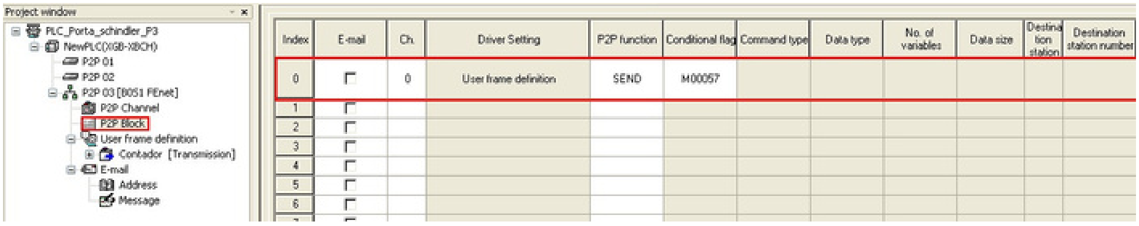
Form: Variable sized variable

☑ Assign memory

Conversion: NONE

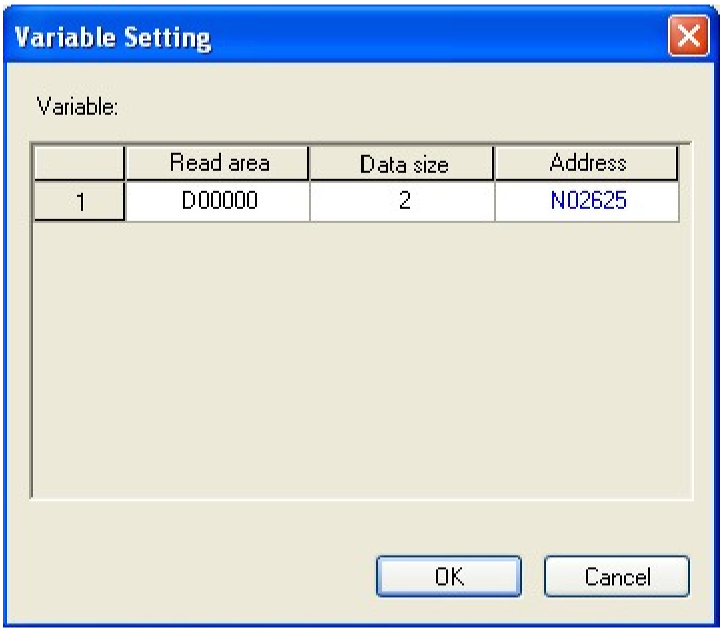
Swap: NONE

24- Double-click on P2P Block and configure the parameters as shown in the screen below:



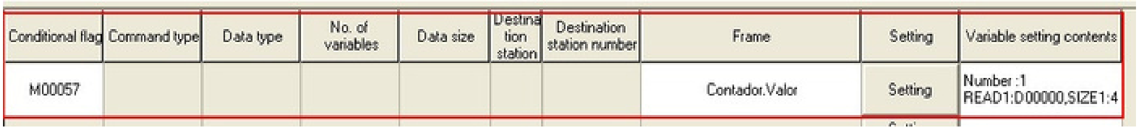
Index	E-mail	Ch.	Driver Setting	P2P function	Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number
0	□	0	User frame definition	SEND	M00057						
1	□										
2	□										
3	□										
4	□										
5	□										
6	□										

Click on setting and configure as shown in the screen below:



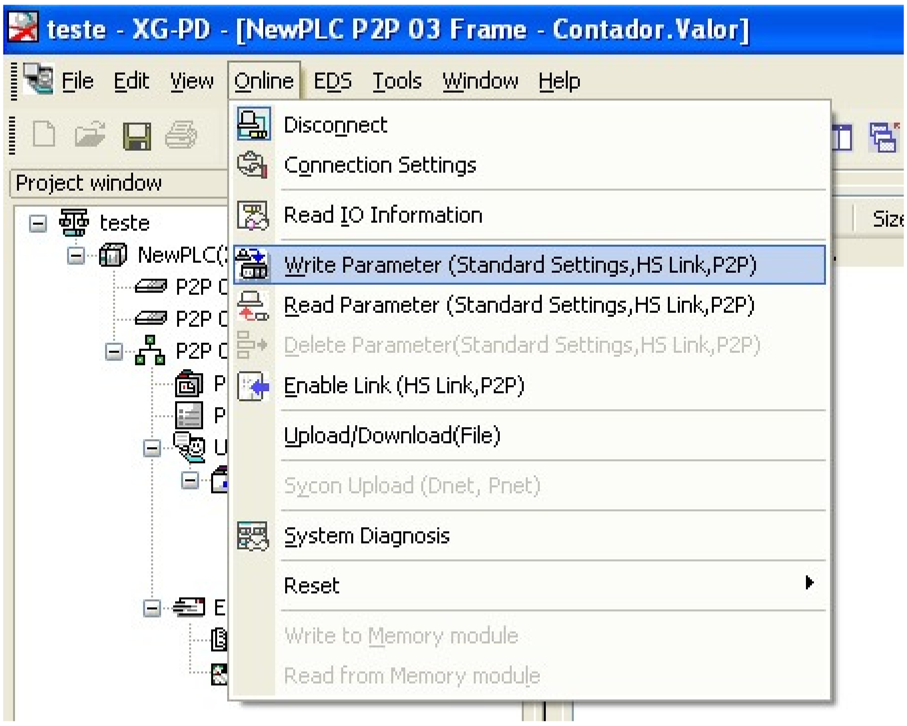
	Read area	Data size	Address
1	D00000	2	N02625

In this example, we have placed the D0 memory in the Read area. This means that the data contained in the D0 memory will be read and sent via Ethernet to the destination computer.

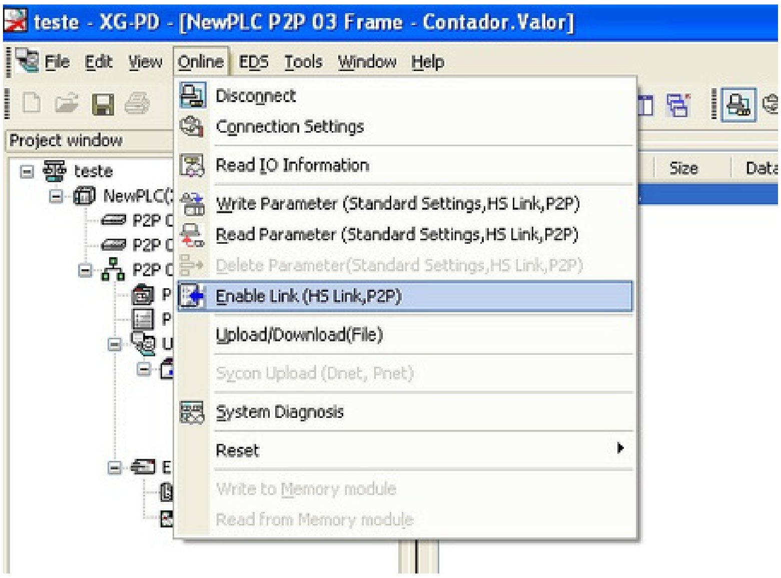


Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number	Frame	Setting	Variable setting contents
M00057							Contador.Valor	Setting	Number :1 READ1:D00000,SIZE1:4

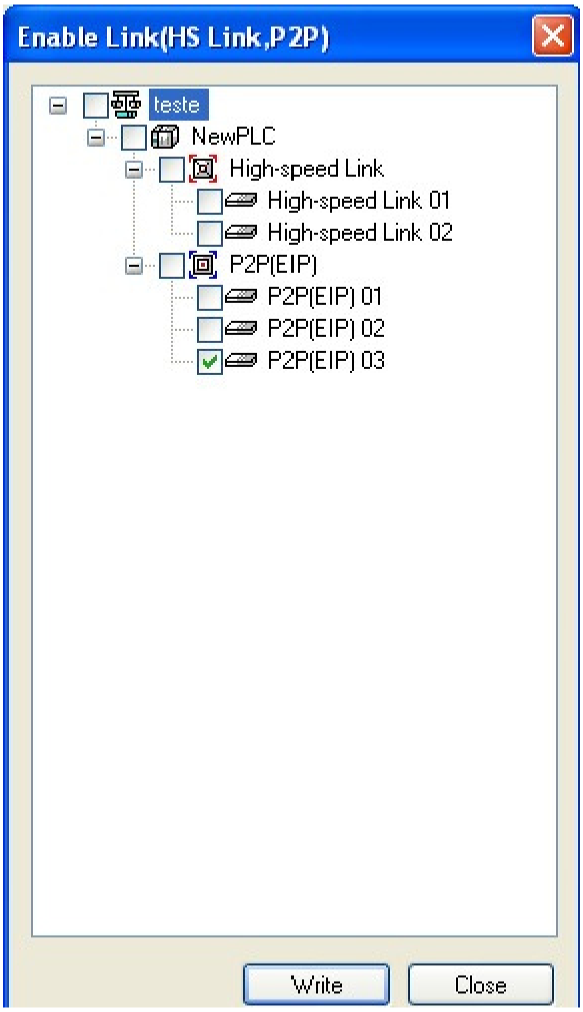
25. Click Online > Write Parameter and then OK:


teste - XG-PD - [NewPLC P2P 03 Frame - Contador.Valor]
File
Edit
View
Online
EDS
Tools
Window
Help
Disconnect
Connection Settings
Read IO Information
Write Parameter (Standard Settings,HS Link,P2P)
Read Parameter (Standard Settings,HS Link,P2P)
Delete Parameter(Standard Settings,HS Link,P2P)
Enable Link (HS Link,P2P)
Upload/Download(File)
Sycon Upload (Dnet, Pnet)
System Diagnosis
Reset
Write to Memory module
Read from Memory module
Project window
teste
NewPLC(
P2P
Size

26- Click em Online > Enable Link:


teste - XG-PD - [NewPLC P2P 03 Frame - Contador.Valor]
File
Edit
View
Online
EDS
Tools
Window
Help
Disconnect
Connection Settings
Read IO Information
Write Parameter (Standard Settings,HS Link,P2P)
Read Parameter (Standard Settings,HS Link,P2P)
Delete Parameter(Standard Settings,HS Link,P2P)
Enable Link (HS Link,P2P)
Upload/Download(File)
Sycon Upload (Dnet, Pnet)
System Diagnosis
Reset
Write to Memory module
Read from Memory module
Project window
teste
NewPLC(
P2P
Size
Data

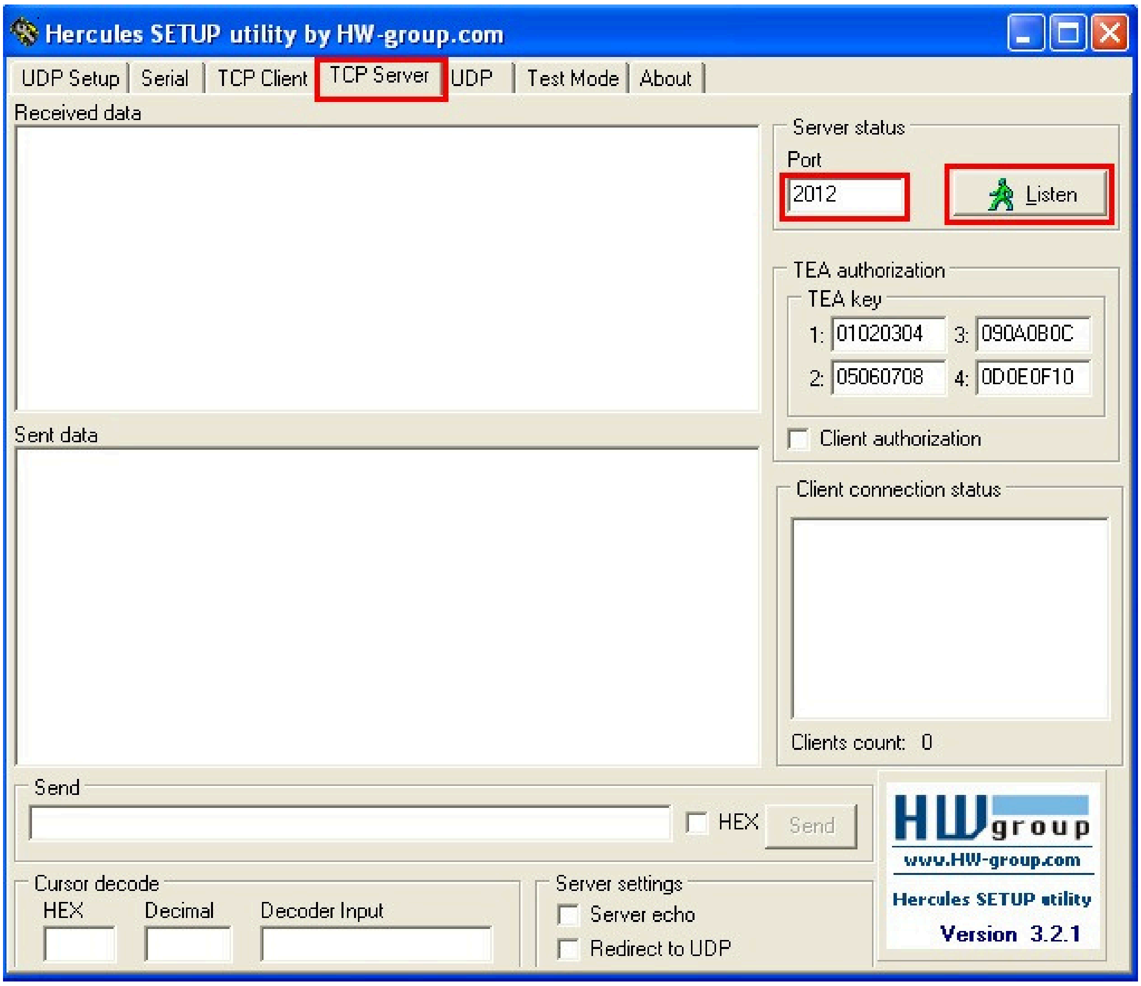
27- As the configuration was done in P2P03, check this option to be enabled and then click on Write > OK > Close:



Enable Link(HS Link,P2P)

```
□ teste
  □ NewPLC
    □ High-speed Link
      □ High-speed Link 01
      □ High-speed Link 02
    □ P2P(EIP)
      □ P2P(EIP) 01
      □ P2P(EIP) 02
      ☑ P2P(EIP) 03
```

28- We can perform a test using a program for Ethernet communication, in this example we use the Hercules Software to read the data sent from the PLC to the Computer. Configure according to the screen below and click on Listen:



Port: 2012

TEA key 1: 01020304 2: 05060708 3: 090A0B0C 4: 0D0E0F10

□ Client authorization

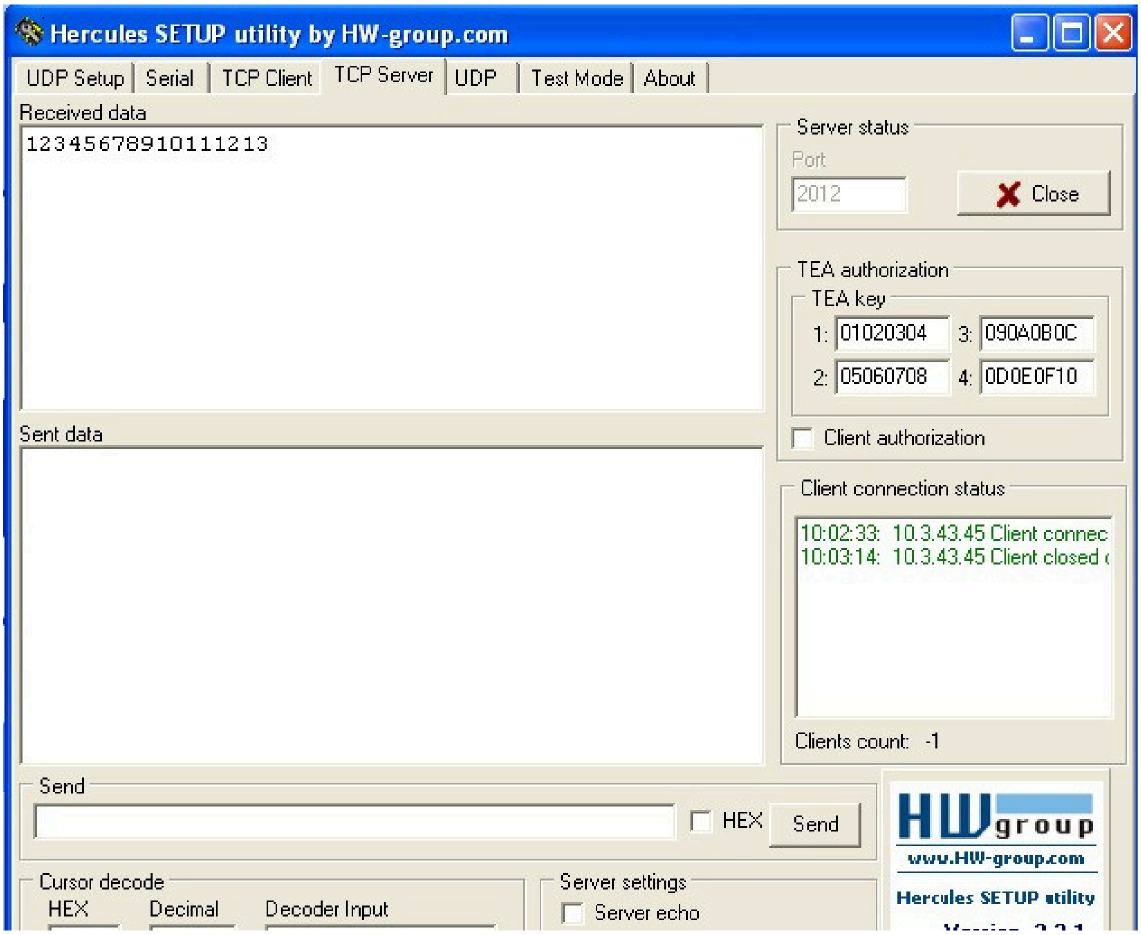
Clients count: 0

□ HEX

□ Server echo

□ Redirect to UDP

29- When memory M57 goes to 1, as previously configured, the data in memory D0 will be sent to the Hercules software as shown in the figure below:



Hercules SETUP utility by HW-group.com

Received data

```
12345678910111213
```

Server status

Port: 2012

TEA authorization

TEA key

1: 01020304 3: 090A0B0C

2: 05060708 4: 0D0E0F10

□ Client authorization

Client connection status

```
10:02:33:  10.3.43.45 Client connec
10:03:14:  10.3.43.45 Client closed c
```

Clients count: -1

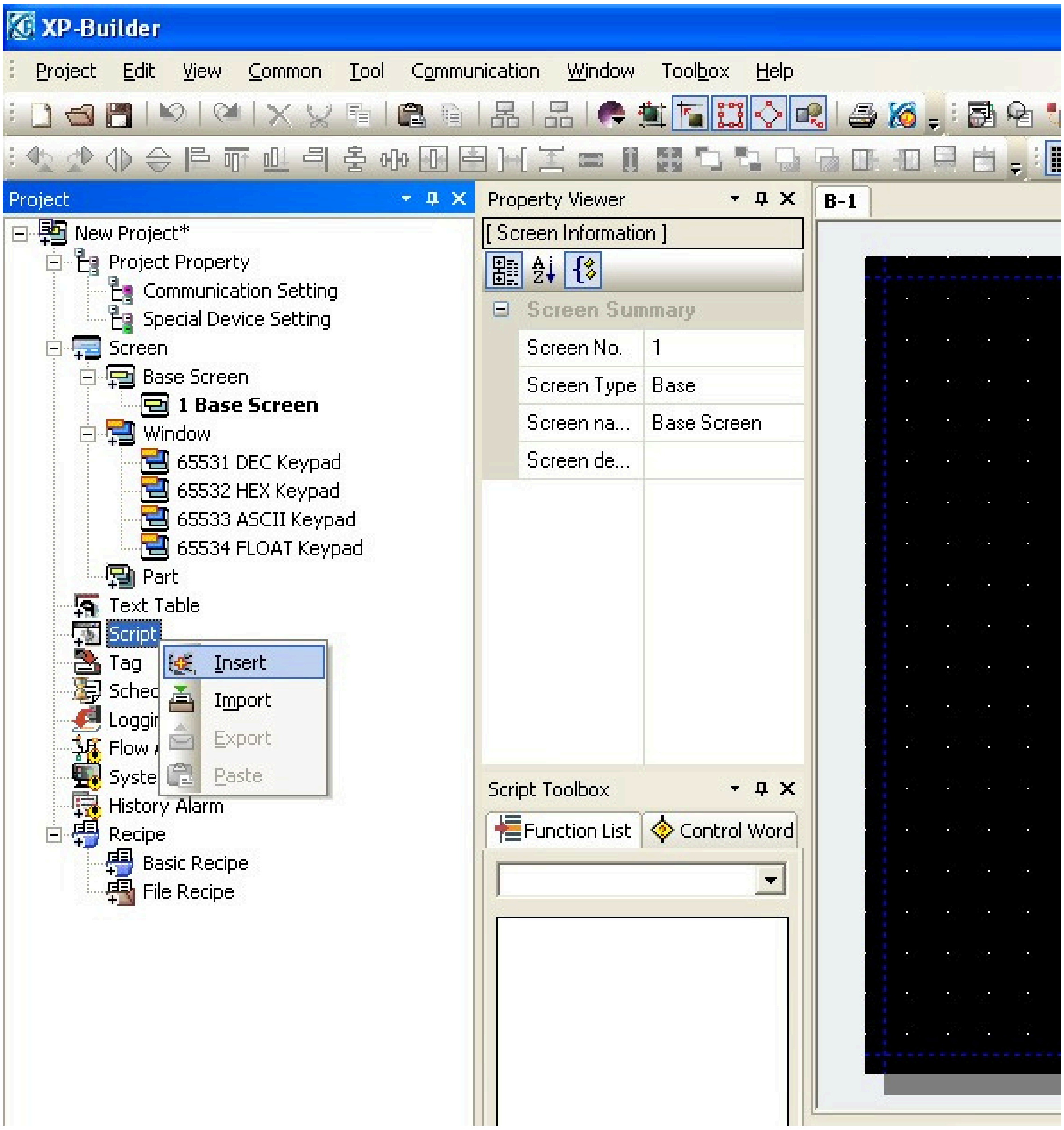
□ HEX

□ Server echo

And

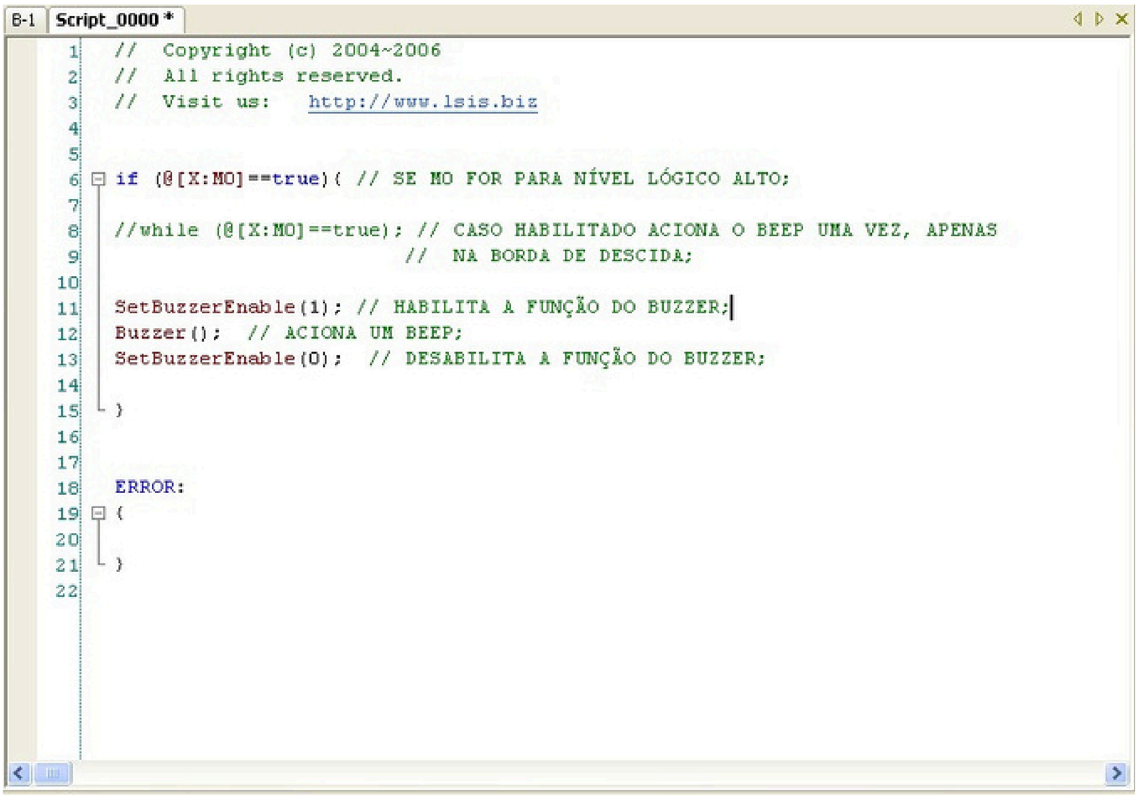
Creating a Script for HMI

1. **In XP-Builder right click on Script > Insert:**



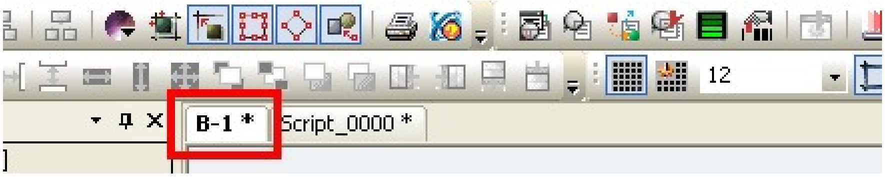
Screen No.	1
Screen Type	Base
Screen na...	Base Screen
Screen de...	

2. **Enter the code as shown in the screen below;**

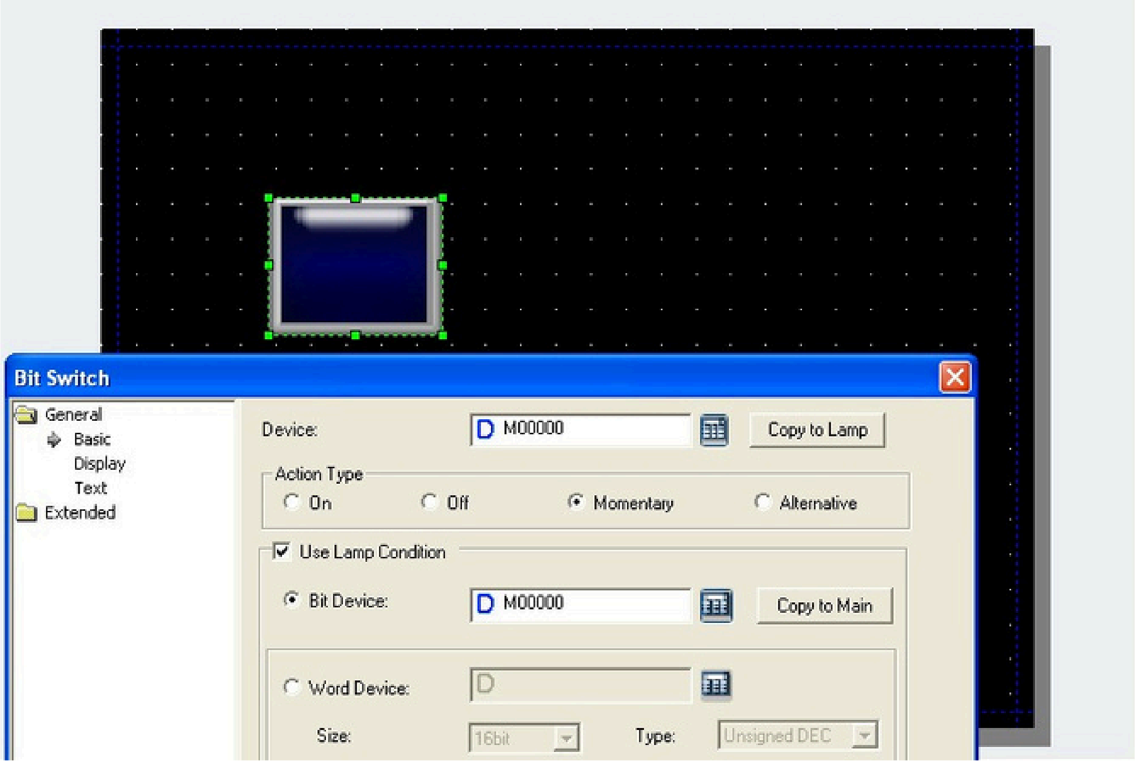


```
//  Copyright (c) 2004~2006
//  All rights reserved.
//  Visit us:   http://www.lsis.biz


if (@[X:MO]==true){ // SE MO FOR PARA NÍVEL LÓGICO ALTO;

//while (@[X:MO]==true); // CASO HABILITADO ACIONA O BEEP UMA VEZ, APENAS
                         //  NA BORDA DE DESCIDA;

SetBuzzerEnable(1); // HABILITA A FUNÇÃO DO BUZZER;
Buzzer();  // ACIONA UM BEEP;
SetBuzzerEnable(0);  // DESABILITA A FUNÇÃO DO BUZZER;

}


ERROR:
{

}
```

3. **Return to screen B-1:**



4. **On the main screen B-1, create a Bit Switch named M0 like the image below:**



Bit Switch

General
Basic
Display
Text
Extended

Device: D M00000 [Copy to Lamp]

Action Type
○ On ○ Off ● Momentary ○ Alternative

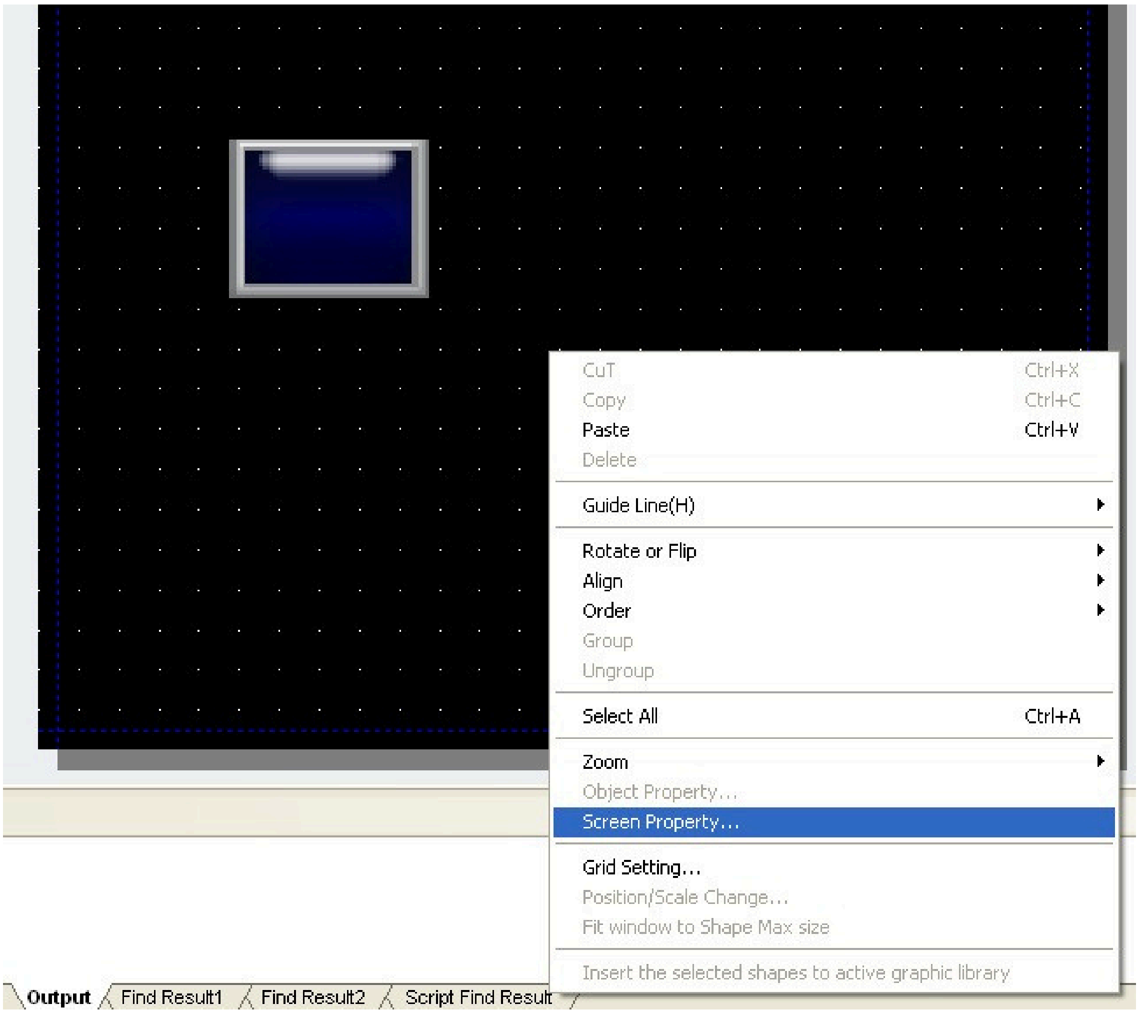
☑ Use Lamp Condition

● Bit Device: D M00000 [Copy to Main]

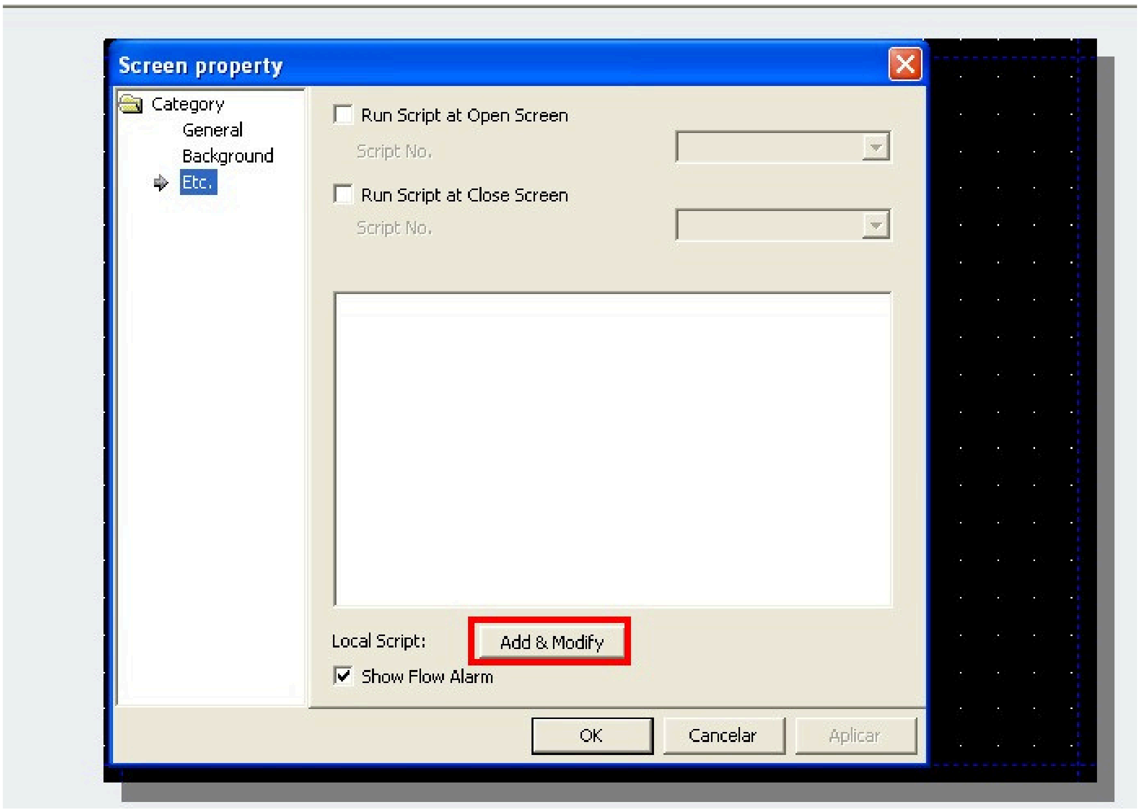
○ Word Device: D

Size: 16bit Type: Unsigned DEC

5. **Right-click on the screen and then click on Screen Property:**



6. Click on Etc. and then on Add & Modify:



□ Run Script at Open Screen

Script No.

□ Run Script at Close Screen

Script No.

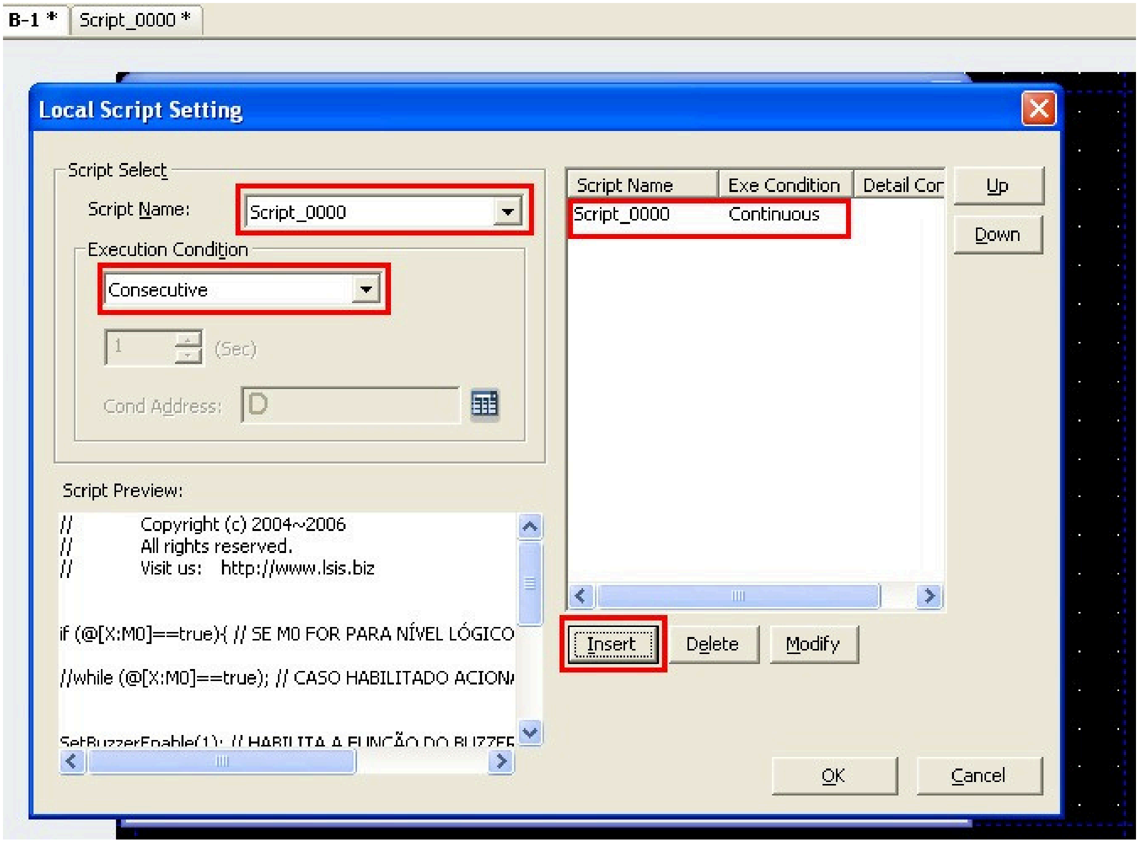
Local Script:

☑ Show Flow Alarm

7. Select the Script created in Script Name; In

8. Execution Condition, select Consecutive; Click on

9. Insert;



Script Name: Script_0000

Execution Condition: Consecutive

1 (Sec)

Cond Address: D

```
//          Copyright (c) 2004~2006
//          All rights reserved.
//          Visit us:    http://www.lsis.biz


if (@[X:M0]==true){ // SE M0 FOR PARA NÍVEL LÓGICO

//while (@[X:M0]==true); // CASO HABILITADO ACION/

SetBuzzerEnable(1); // HABILITA A FUNÇÃO DO BUZZER
```

Script Name	Exe Condition	Detail Cor
Script_0000	Continuous	

10. Click OK and then OK again;

11. Transfer the program to the HMI;

12.

Now, every time M0 goes to logic level 1, the Beep function will be activated.

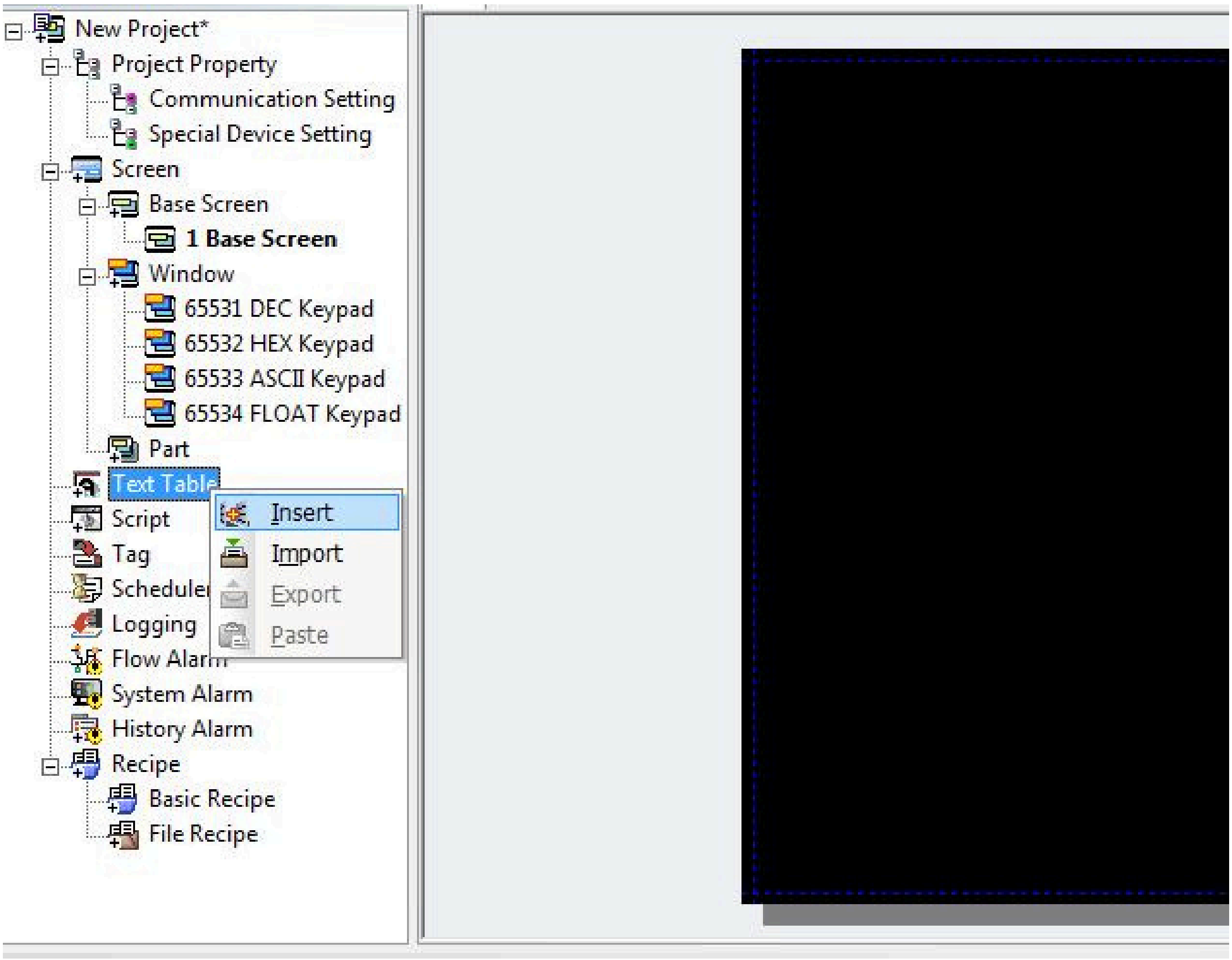
Syntax for memories in programming:

Device	Status Examples	Examples of Use
X : BIT device	@[X:No: *1]	@[X:0:P001], @[X:P001]
S : SHORT(16bit) device	@[S:No: *1]	@[S:0:P000], @[S:2:#1:P000]
L : INT (32 bit) device	@[L:No: *1]	@[L:0:P000], @[L:P000]
W : UNSIGNED SHORT(16 bit)	@[W:No: *1]	@[W:0:P000], @[W:1:#3:P000]
D : UNSIGNED INT(32 bit)	@[D:No: *1]	@[D:0:P000], @[D:#1:P000]
F : FLOAT device	@[F:No: *1]	@[F:0:P000], @[F:#1:P000]

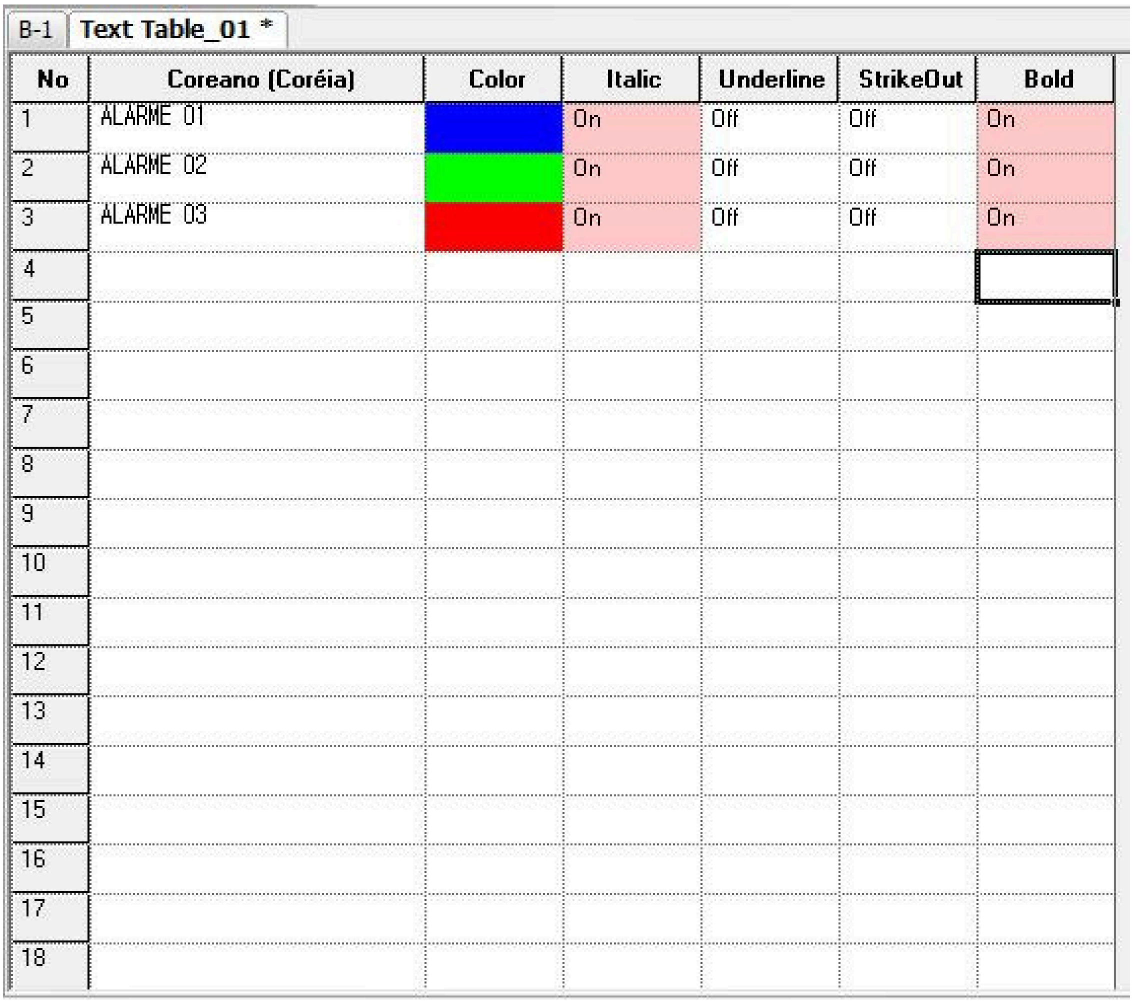
And.

Alarm History and POP-UP

1- **In XP-Builder, right-click on Text Table and then on Insert:**



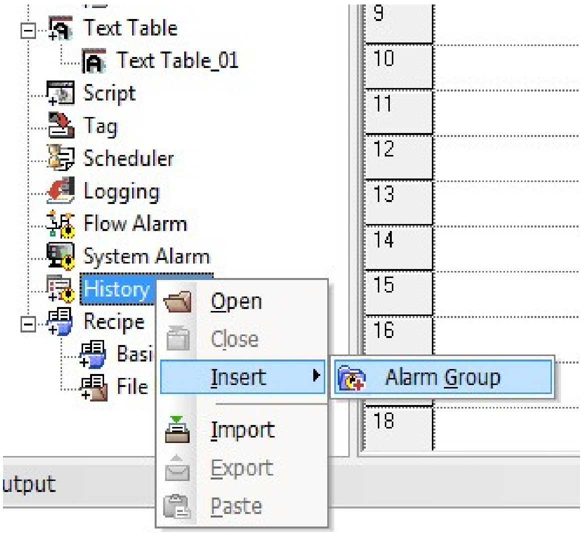
2- We will have the following table that can be configured as shown in the image below:



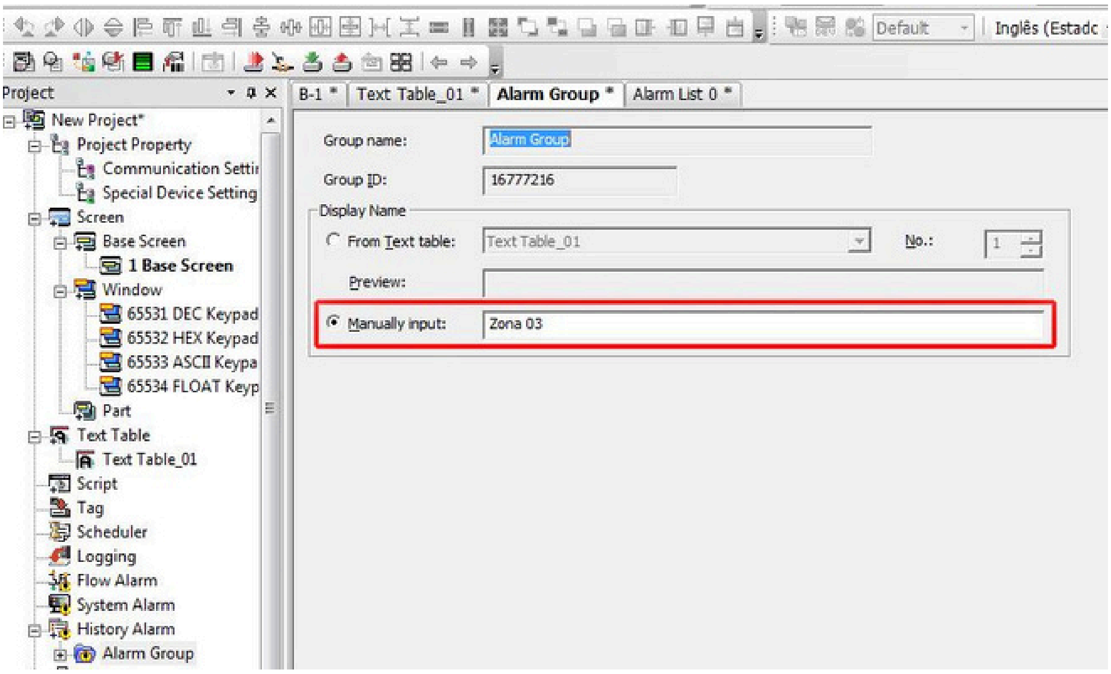
No	Coreano (Coréia)	Color	Italic	Underline	StrikeOut	Bold
1	ALARME 01	🟦	On	Off	Off	On
2	ALARME 02	🟩	On	Off	Off	On
3	ALARME 03	🟥	On	Off	Off	On
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						

The description in the second column “Alarm 01” is the message that the user will receive if the condition of alarm 1, which we will see later, occurs.

3- **Right click on History Alarm > Insert > Alarm Group:**



4- **We can write on the Alarm Group screen the group to which this alarm belongs, in this case Zone 03:**



Group name: Alarm Group

Group ID: 16777216

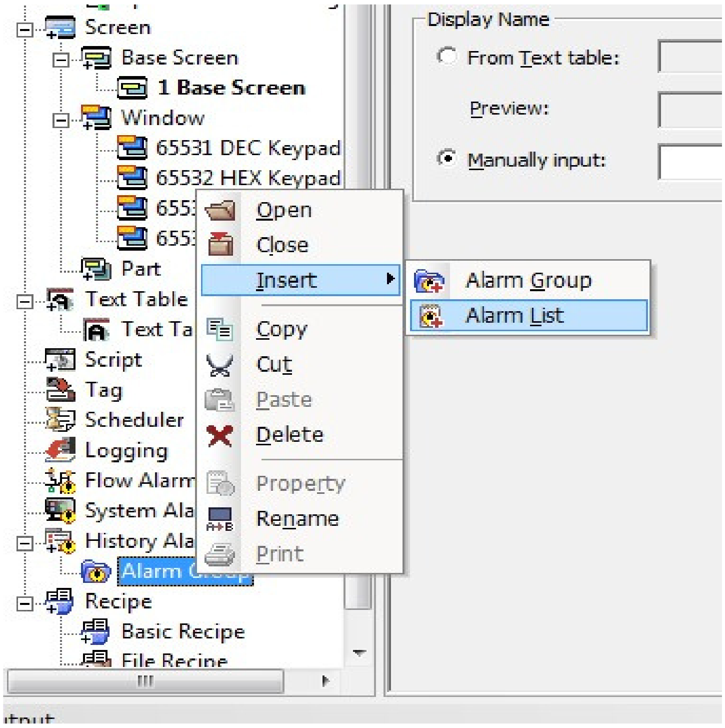
Display Name

○ From Text table: Text Table_01 No.: 1

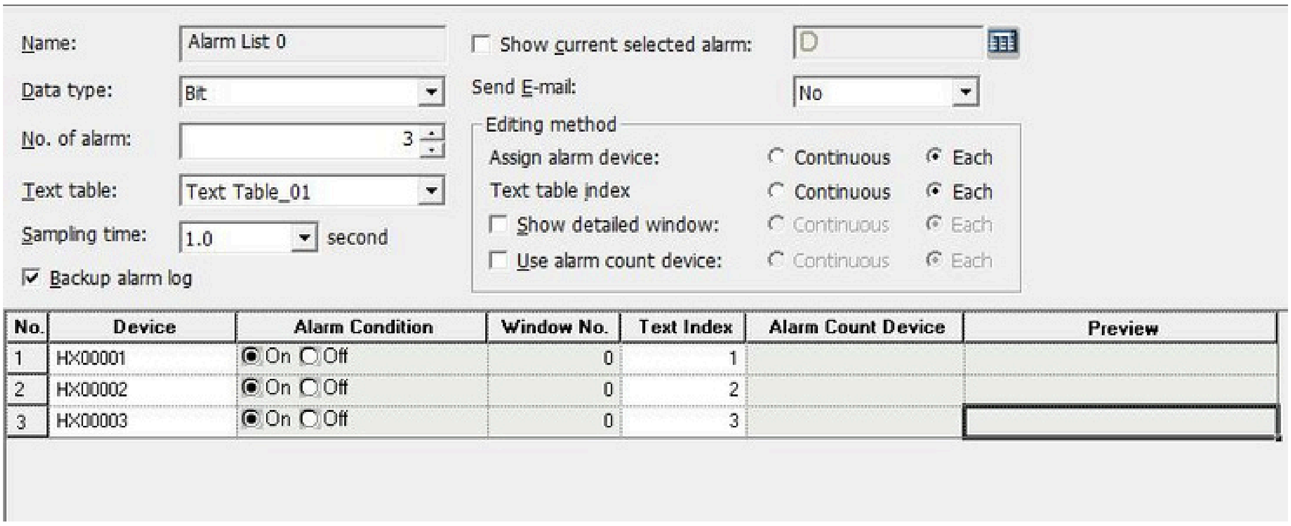
Preview:

● Manually input: Zona 03

5- **Then right-click on Alarm Group > Insert > Alarm List:**



6- **Configure the data as shown in the screen below:**



Name: Alarm List 0

Data type: Bit

No. of alarm: 3

Text table: Text Table_01

Sampling time: 1.0 second

☑ Backup alarm log

□ Show current selected alarm: D

Send E-mail: No

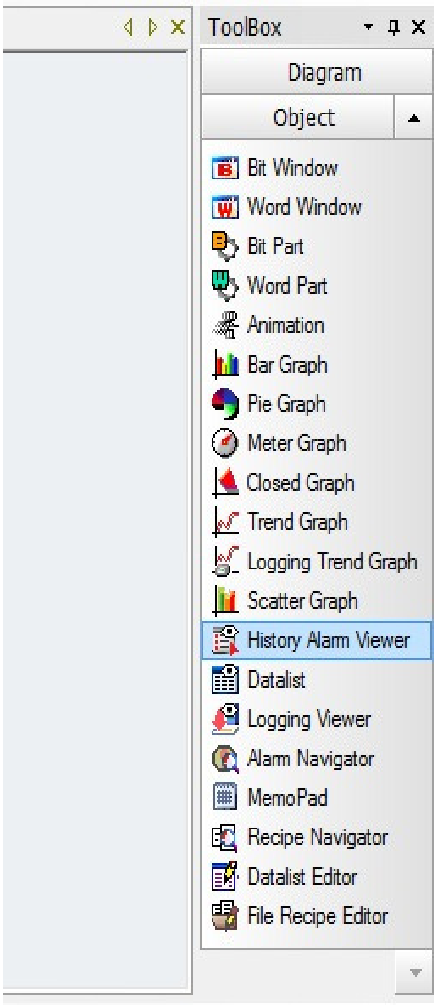
Editing method

Assign alarm device:	○ Continuous	● Each
Text table index	○ Continuous	● Each
□ Show detailed window:	○ Continuous	● Each
□ Use alarm count device:	○ Continuous	● Each

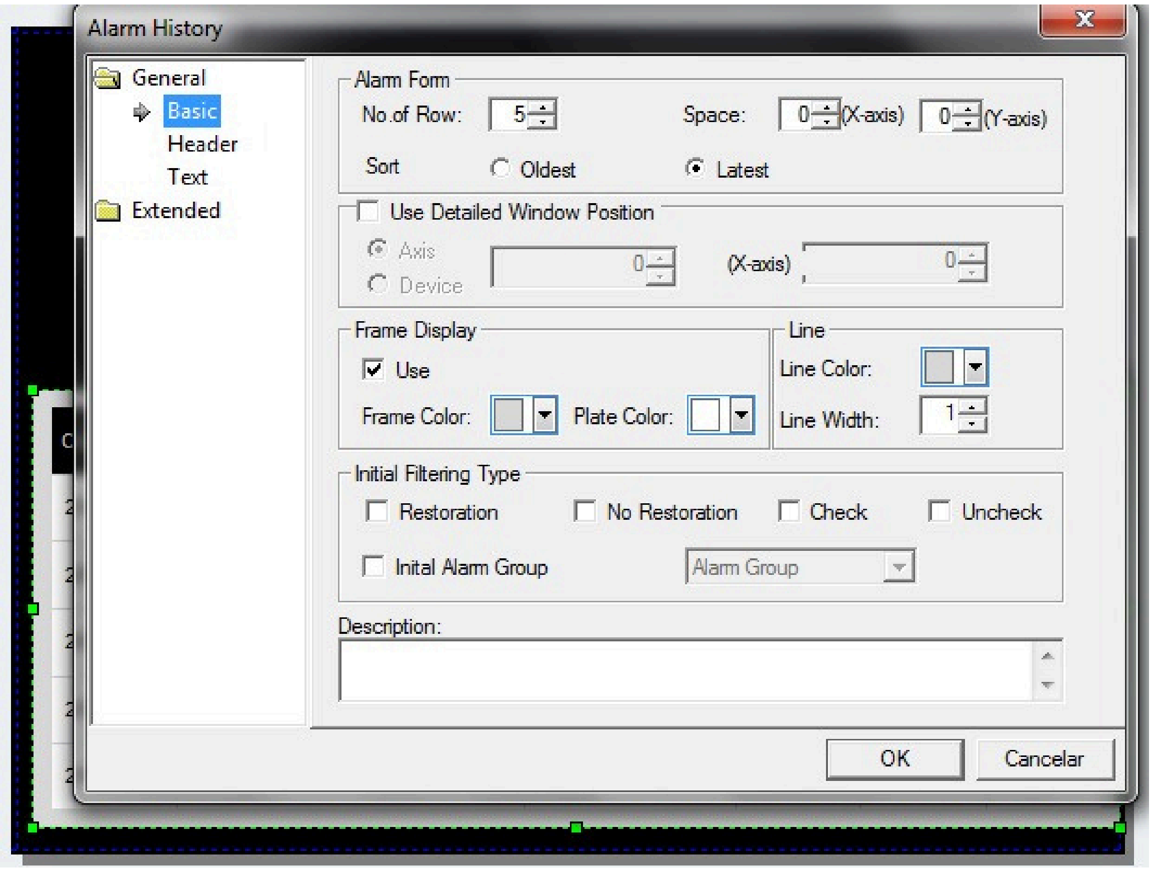
No.	Device	Alarm Condition	Window No.	Text Index	Alarm Count Device	Preview
1	HX00001	● On ○ Off	0	1		
2	HX00002	● On ○ Off	0	2		
3	HX00003	● On ○ Off	0	3		

HX1, HX2 and HX3 are internal memories of the HMI. If we were to use memories from the LS PLC, we could use memories M1, M2 and M3, for example. In Alarm Condition we define when the alarm action will occur. In this example, when any of these memories goes to a high logic level (On), an alarm warning will be written in a table that we will create next.

7- Click on History Alarm Viewer to create the table in the Base Screen:



8- **Click on the screen and drag to create the table and then click OK:**



Alarm History

Alarm Form

No.of Row: 5

Space: 0 (X-axis) 0 (Y-axis)

Sort ○ Oldest ● Latest

□ Use Detailed Window Position

● Axis 0 (X-axis) 0

○ Device

Frame Display

☑ Use

Frame Color: Plate Color:

Line

Line Color:

Line Width: 1

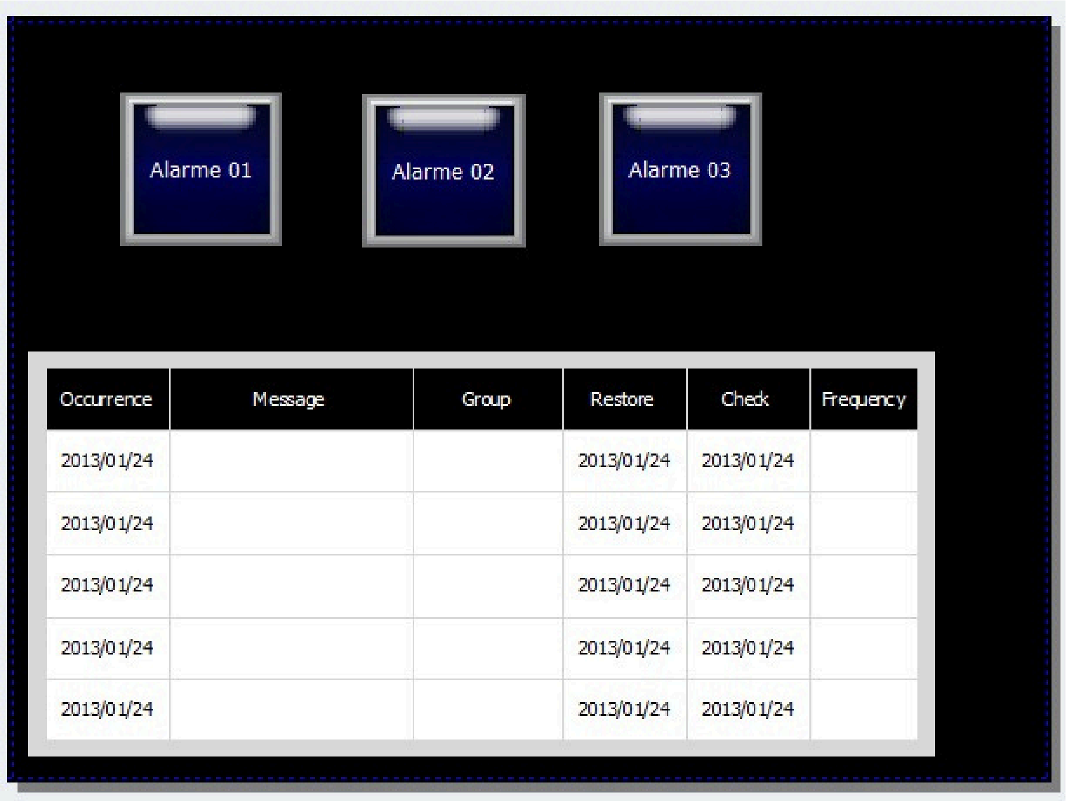
Initial Filtering Type

□ Restoration □ No Restoration □ Check □ Uncheck

□ Inital Alarm Group Alarm Group

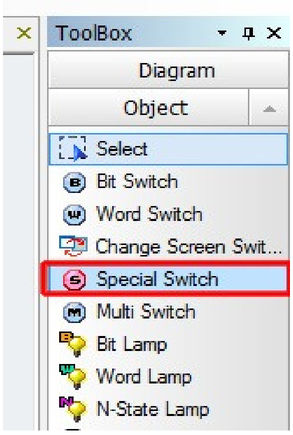
Description:

9- Create three buttons (Bit Switch - Momentary) as HX1, HX2 and HX3 respectively on the screen that will serve to simulate the alarms. Every time one of these buttons is pressed, it will be recorded in the table.

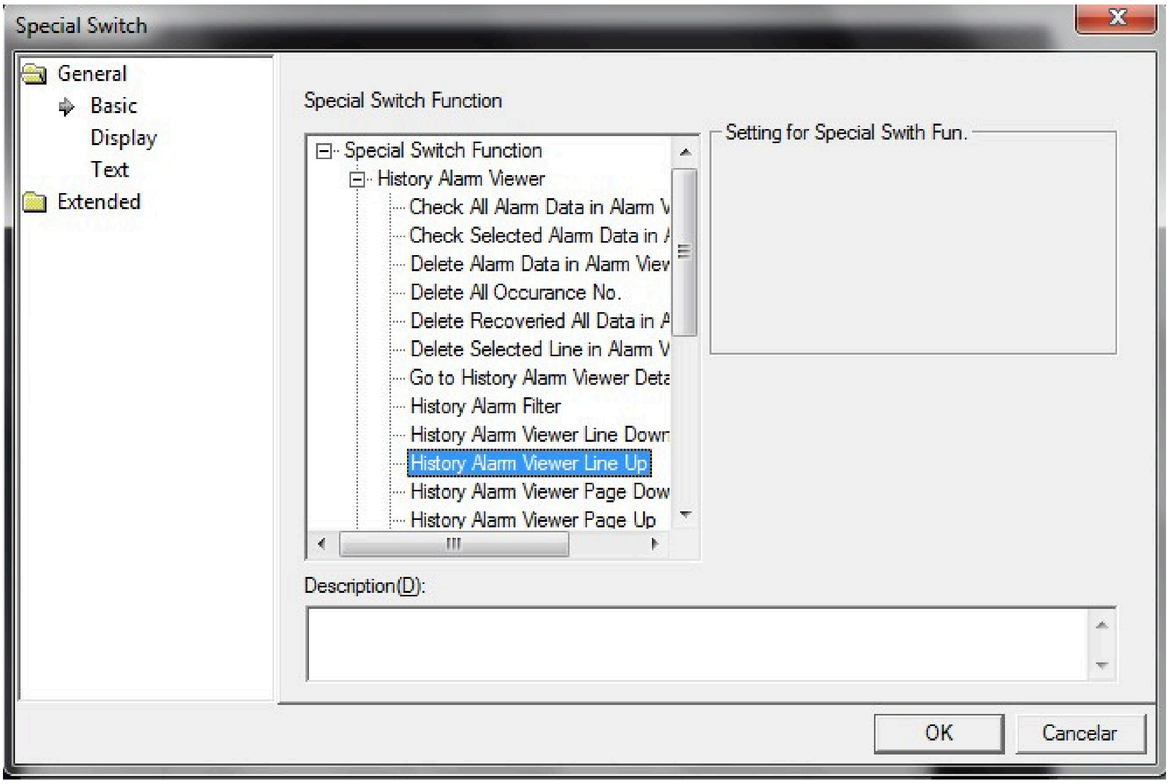


Occurrence	Message	Group	Restore	Check	Frequency
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	

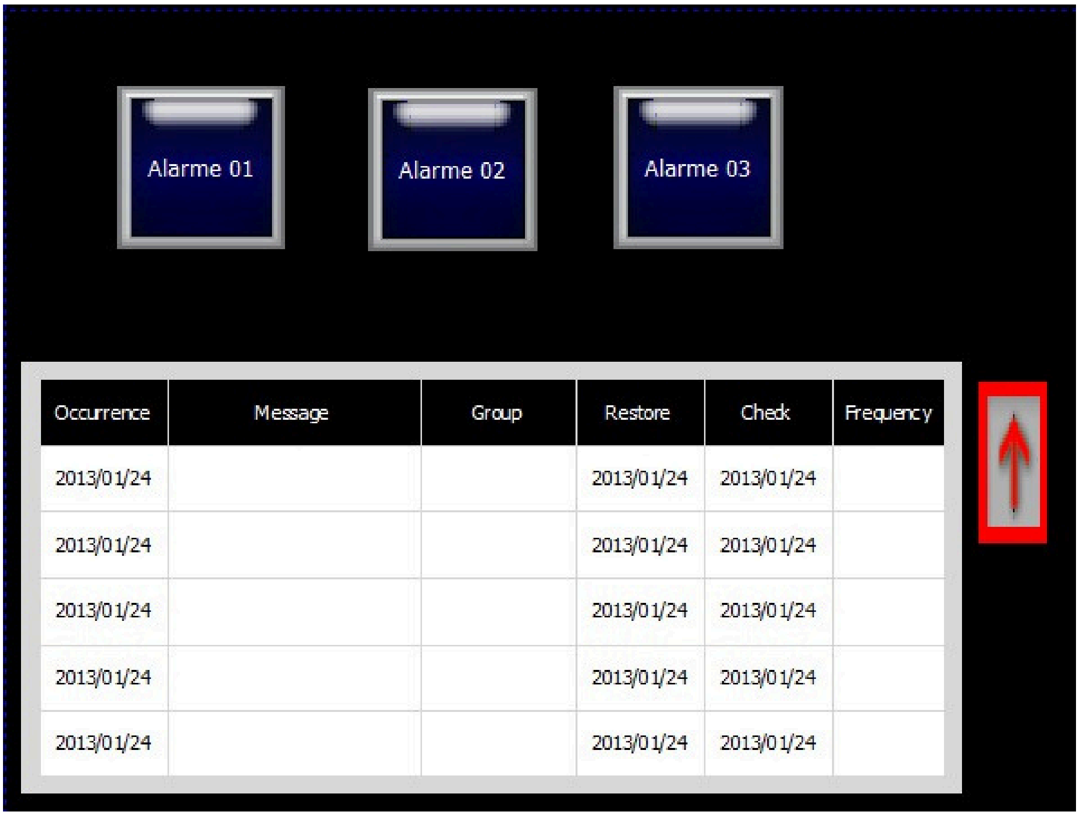
10- We can also create a scroll on the screen to check all occurrences, click on Special Switch:



11- Create a button on the screen next to the table and in the window that will open, select the History Alarm Viewer Line Up option within
History Alarm Viewer:

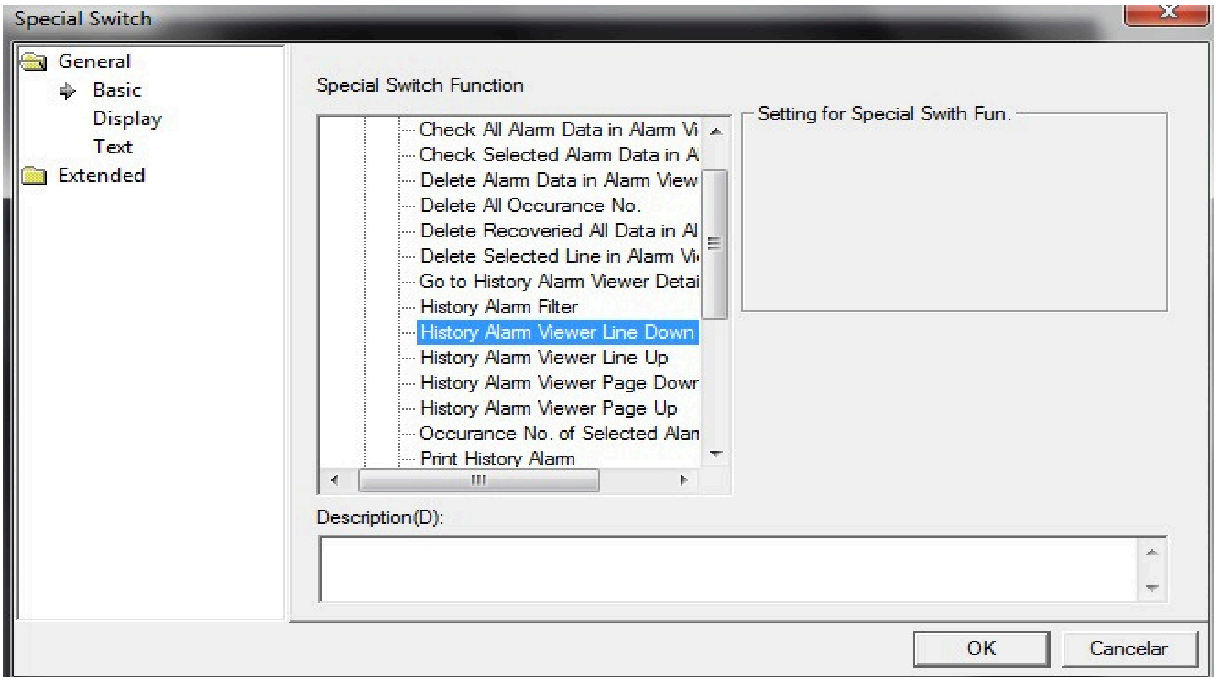

Special Switch
General
Basic
Display
Text
Extended
Special Switch Function
Special Switch Function
History Alarm Viewer
Check All Alarm Data in Alarm V
Check Selected Alarm Data in A
Delete Alarm Data in Alarm View
Delete All Occurance No.
Delete Recoveried All Data in A
Delete Selected Line in Alarm V
Go to History Alarm Viewer Deta
History Alarm Filter
History Alarm Viewer Line Down
History Alarm Viewer Line Up
History Alarm Viewer Page Dow
History Alarm Viewer Page Up
Setting for Special Swith Fun.
Description(D):
OK
Cancelar

12- After creating the button, it will look similar to the screen below:

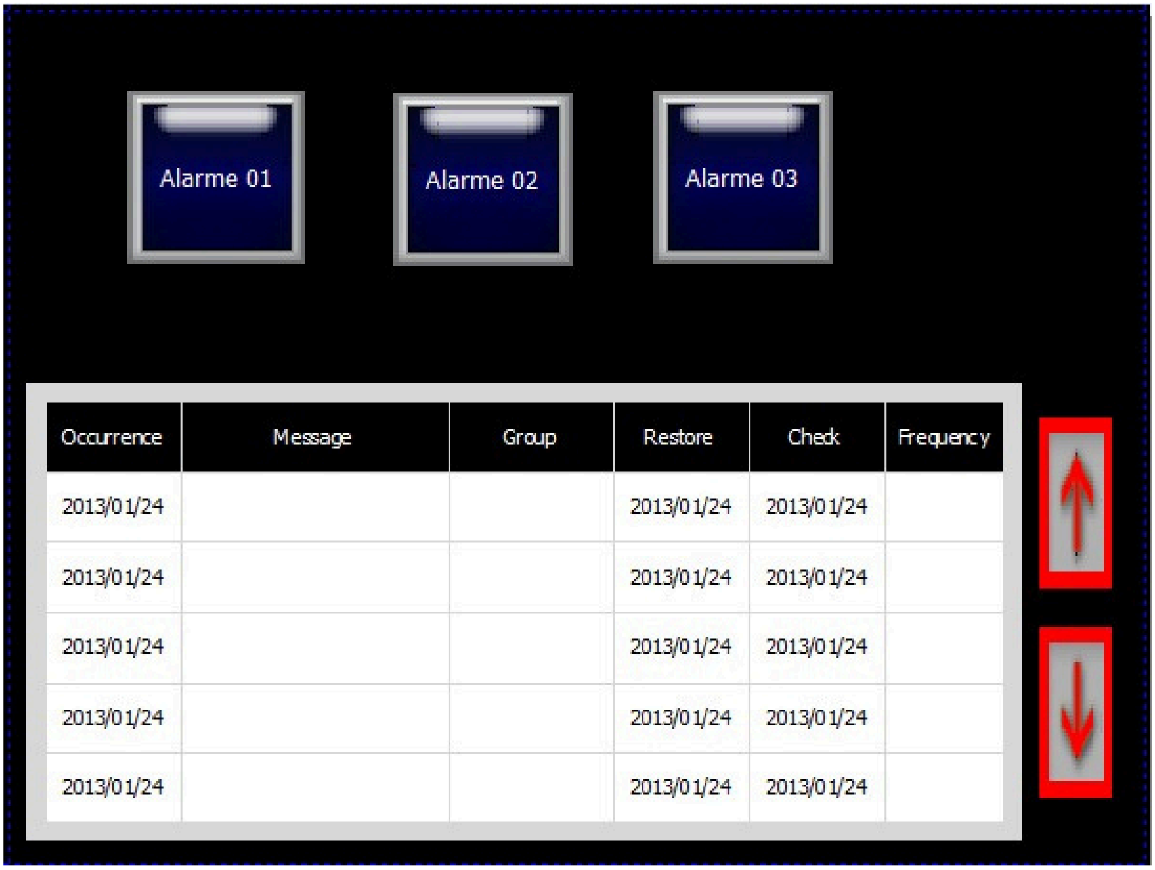


Occurrence	Message	Group	Restore	Check	Frequency
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	

13- Create another button by selecting the button already created and using CTRL C to copy it and CTRL V to paste the new button. Double-click on the second button and change it to the History Alarm Viewer Line Down option:

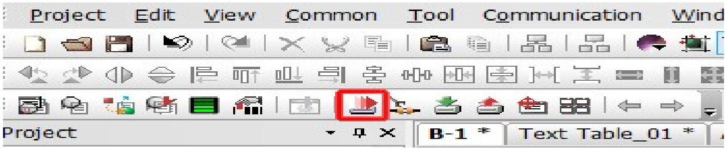

Special Switch
General
Basic
Display
Text
Extended
Special Switch Function
Check All Alarm Data in Alarm Vi
Check Selected Alarm Data in A
Delete Alarm Data in Alarm View
Delete All Occurance No.
Delete Recoveried All Data in Al
Delete Selected Line in Alarm Vi
Go to History Alarm Viewer Detai
History Alarm Filter
History Alarm Viewer Line Down
History Alarm Viewer Line Up
History Alarm Viewer Page Dowr
History Alarm Viewer Page Up
Occurance No. of Selected Alan
Print History Alarm
Setting for Special Swith Fun.
Description(D):
OK
Cancelar

14- The screen will look something like this:

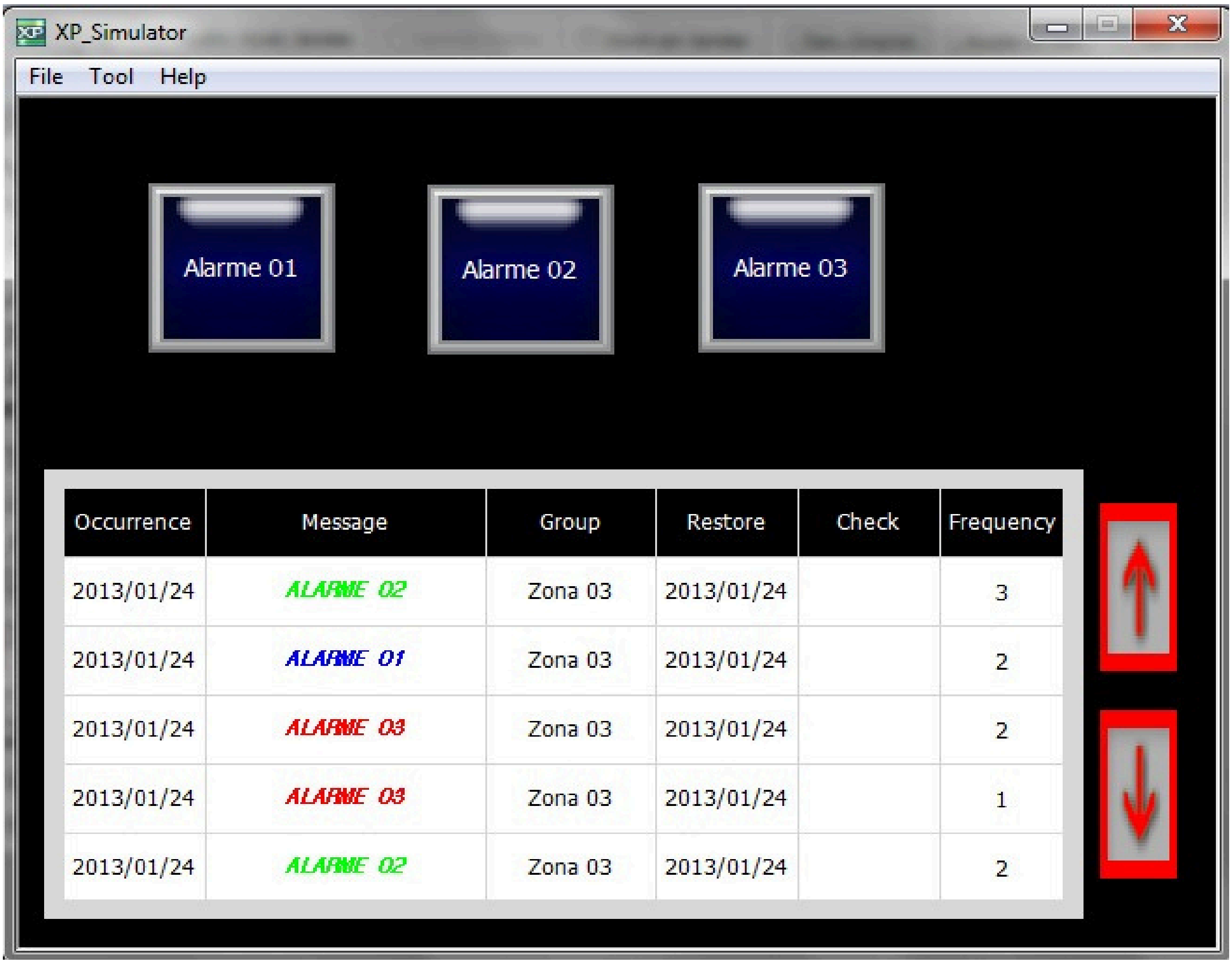


Occurrence	Message	Group	Restore	Check	Frequency
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	
2013/01/24			2013/01/24	2013/01/24	

15- Now we can click on simulation mode to perform a test:



16- By pressing the buttons we notice that they are sequentially recorded in the table.



Occurrence	Message	Group	Restore	Check	Frequency
2013/01/24	ALARME 02	Zona 03	2013/01/24		3
2013/01/24	ALARME 01	Zona 03	2013/01/24		2
2013/01/24	ALARME 03	Zona 03	2013/01/24		2
2013/01/24	ALARME 03	Zona 03	2013/01/24		1
2013/01/24	ALARME 02	Zona 03	2013/01/24		2

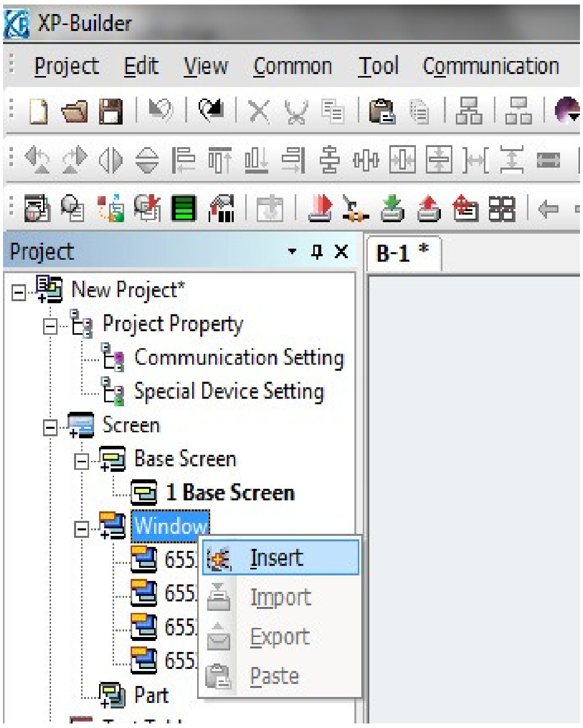
To see all the alarms that have occurred, we can use the two buttons created on the right side of the table, which have the function of scrolling the screen down or up.

Creating POP-UP function

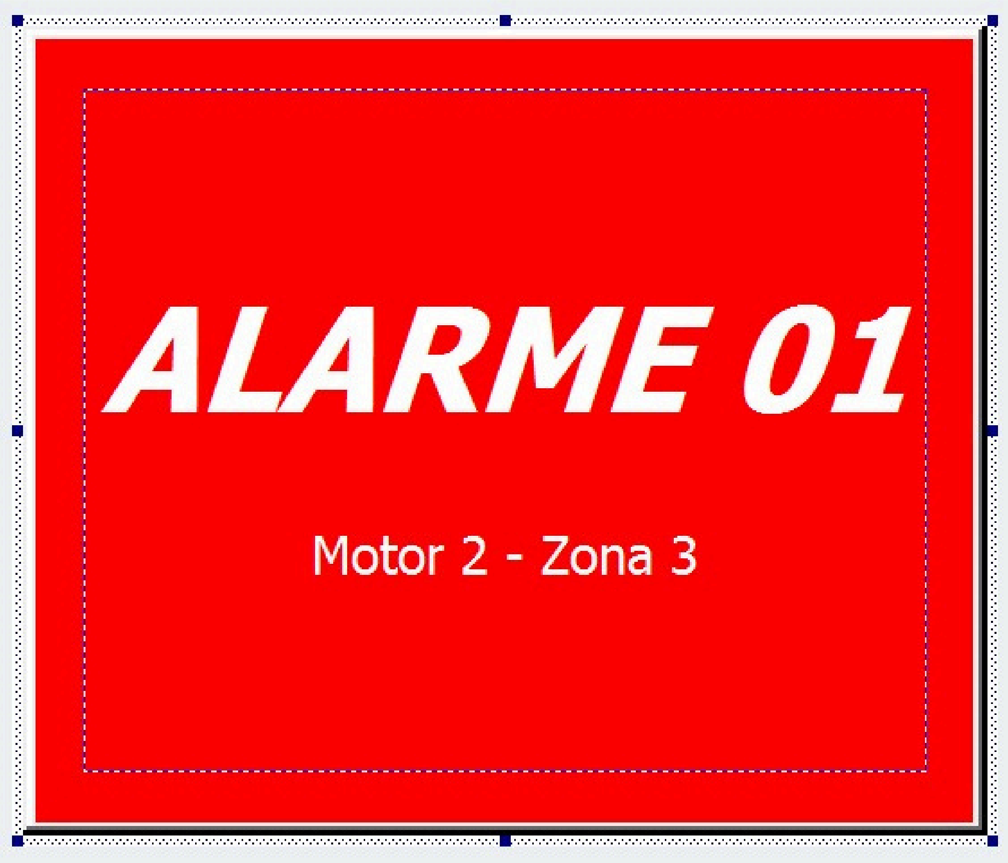
We can also use a POP-UP function on the screen for better viewing of the alarm.

17- Creating another program just to demonstrate the POP-UP screen:

18- Right-click on Windows and then on the Insert option:



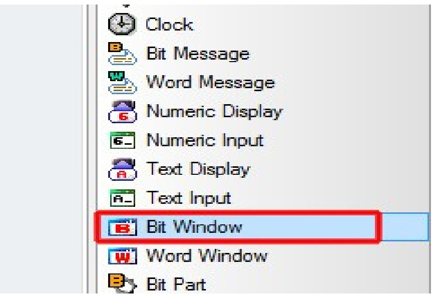
A screen will open so that we can configure the POP-UP design. We can right-click on the screen and then click on the Screen Property option to change the screen color and we can also insert texts corresponding to the alarm.


ALARME 01

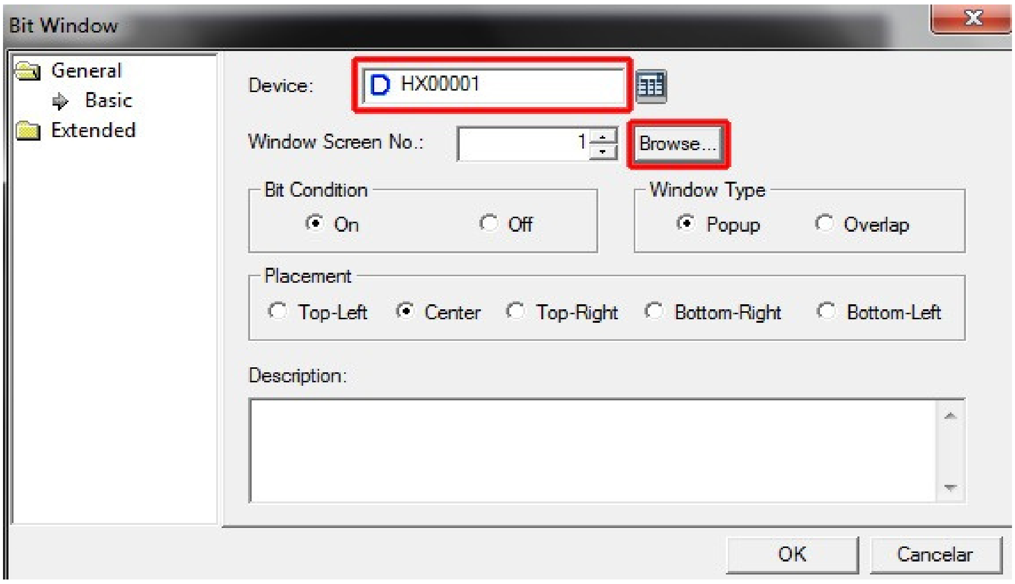
Motor 2 - Zona 3

19- With the POP-UP created, let's now create the Bit Windows function. We need to create this function on the screen where we want the POP-UP to open. If you want to open the alarm POP-UP on all screens, we can use the CTRL C and CTRL V commands to copy it to all screens. Only the screen that contains the Bit Windows function will call the POP-UP.

20- Go back to the base screen and click on the Bit Windows object, drag it on the base screen to create it.



21- Configure the memory corresponding to the alarm. In this case, the HX1 memory corresponding to alarm 01 was configured. However, it could be any memory configured for the alarm in the PLC. Click on Browser and locate the POP-UP screen that was designed previously.


Bit Window

General
Basic
Extended

Device: D HX00001

Window Screen No.: 1 Browse...

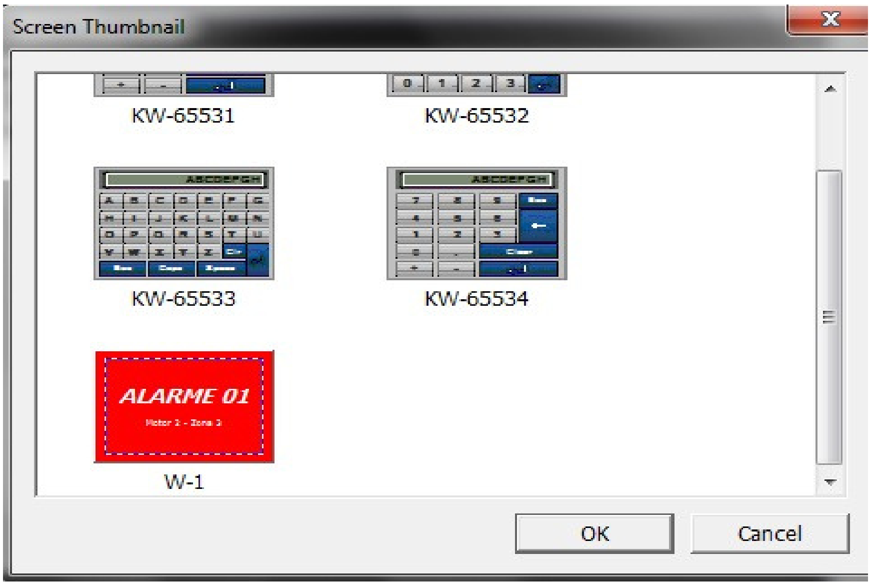
Bit Condition: ● On ○ Off

Window Type: ● Popup ○ Overlap

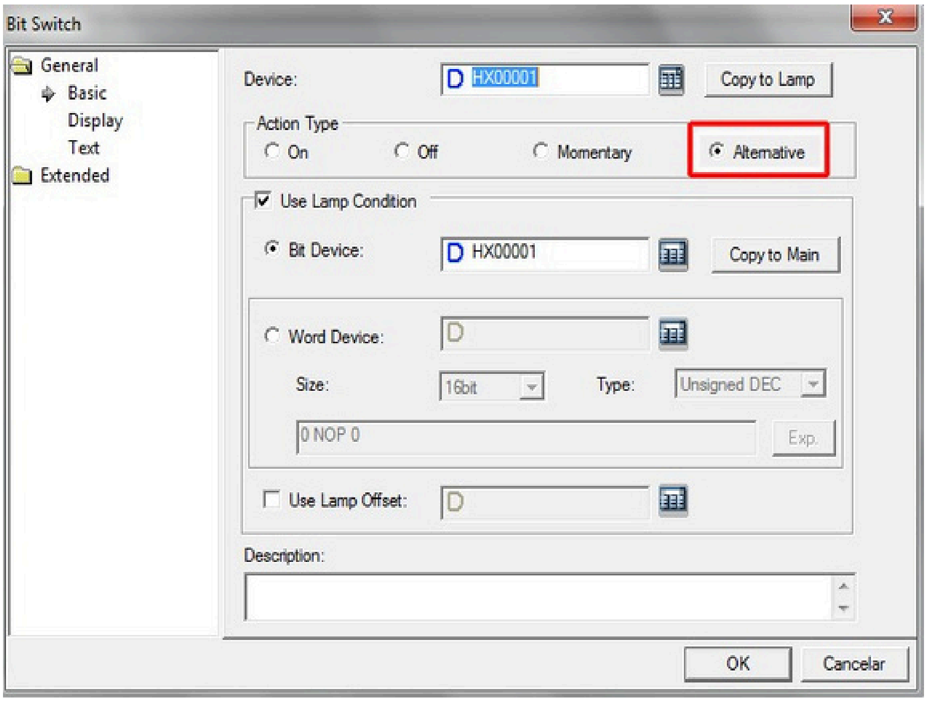
Placement: ○ Top-Left ● Center ○ Top-Right ○ Bottom-Right ○ Bottom-Left

Description:

OK Cancelar



22- To test the program, create a Bit Switch button on the screen as Alternative and memory HX1:



Device: D HX00001

Action Type: ○ On ○ Off ○ Momentary ● Alternative

☑ Use Lamp Condition

● Bit Device: D HX00001

○ Word Device: D

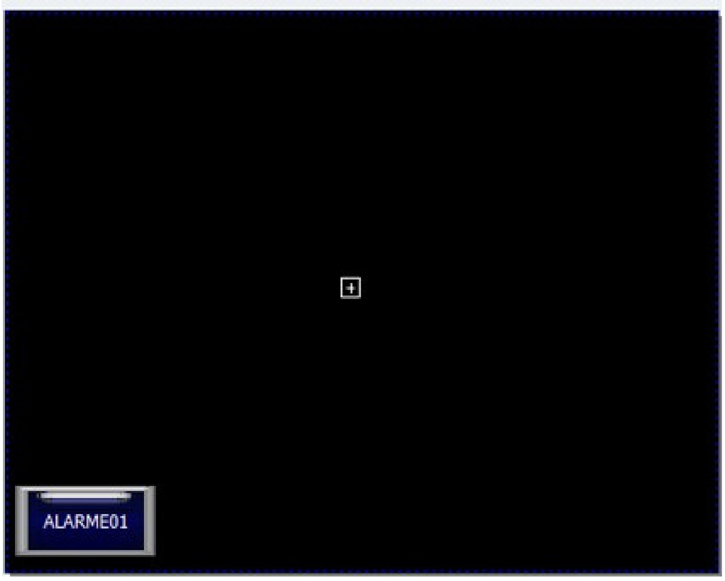
Size: 16bit Type: Unsigned DEC

0 NOP 0

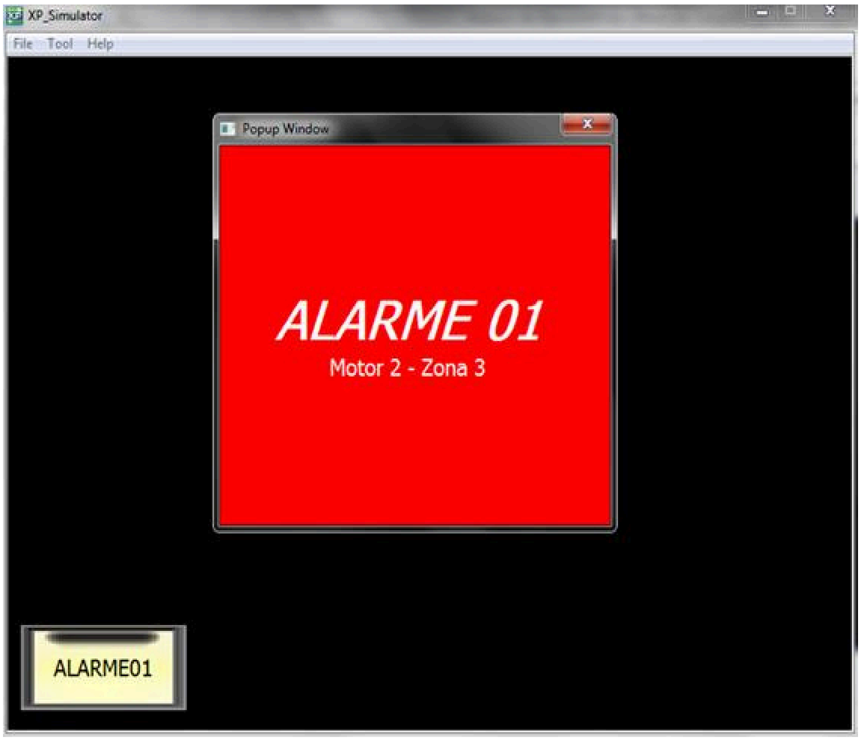
□ Use Lamp Offset: D

Description:

And.



23- We can simulate the program to see the result:



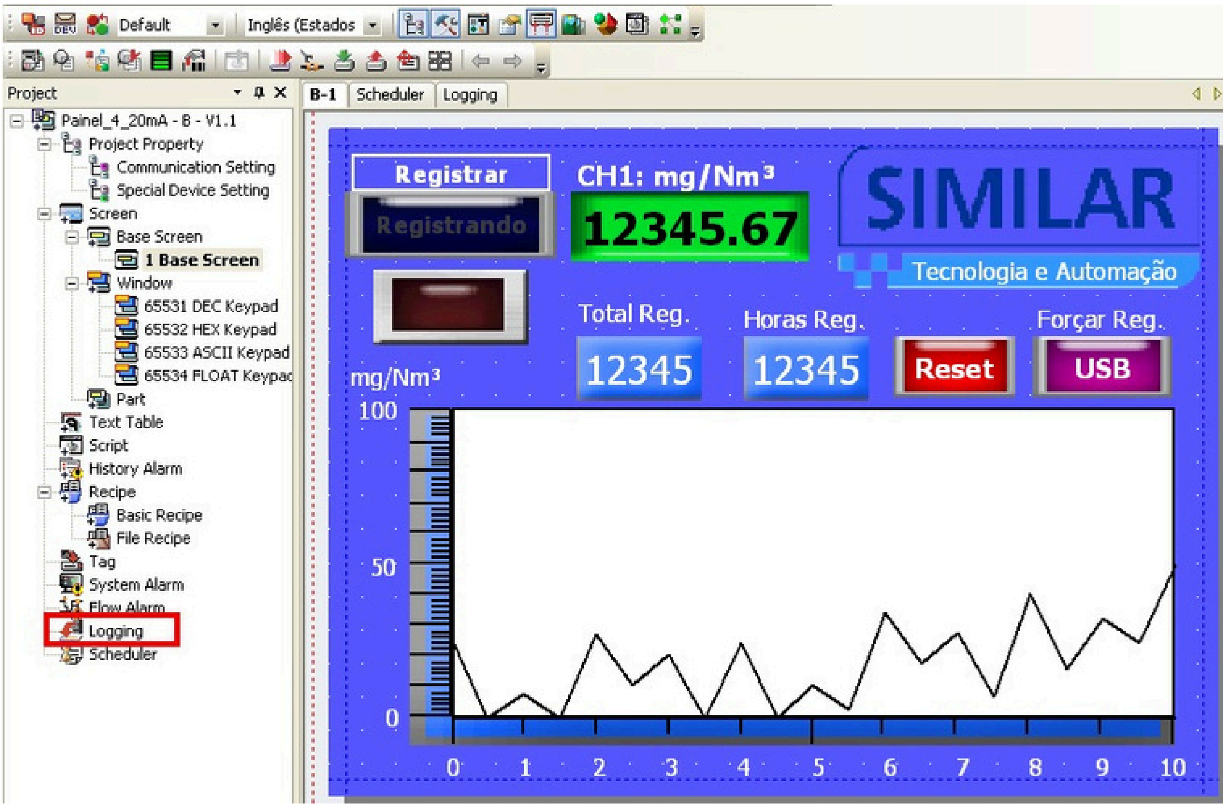
And.

Loggin with sending of E-mail

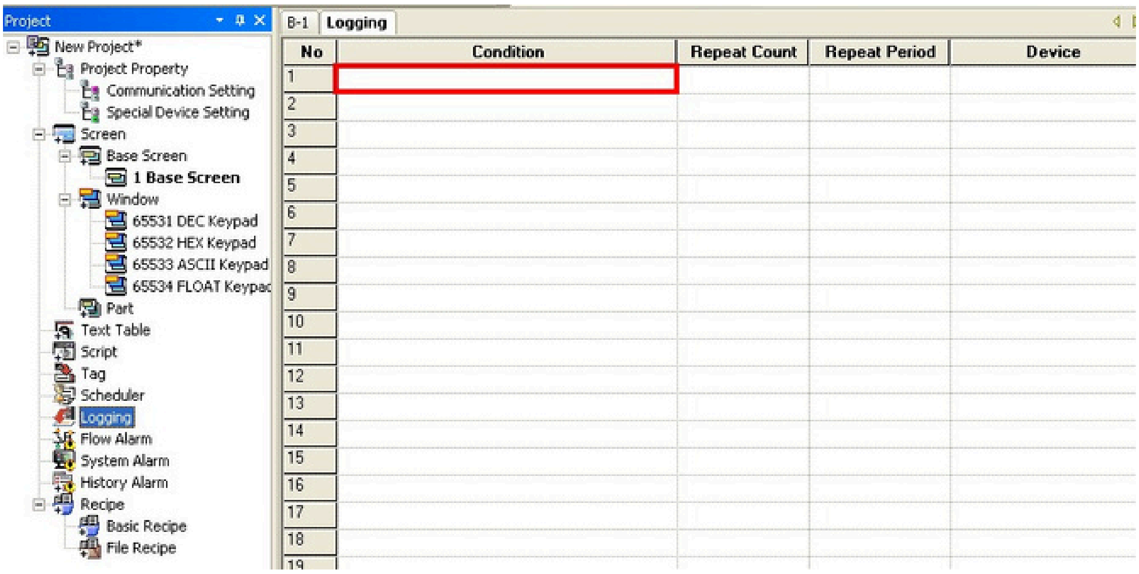
This XP-Builder function allows you to perform data or bit backups from the HMI's internal memory or even from the PLC's memory. The information from the acquisitions made is first recorded in the HMI's internal memory and then sent to an external area, such as a CF Card, Pendrive or Email, configurable depending on the HMI model.

DATA BACKUP CONFIGURATION

- In XP-Builder, double-click on Logging in the column on the left:

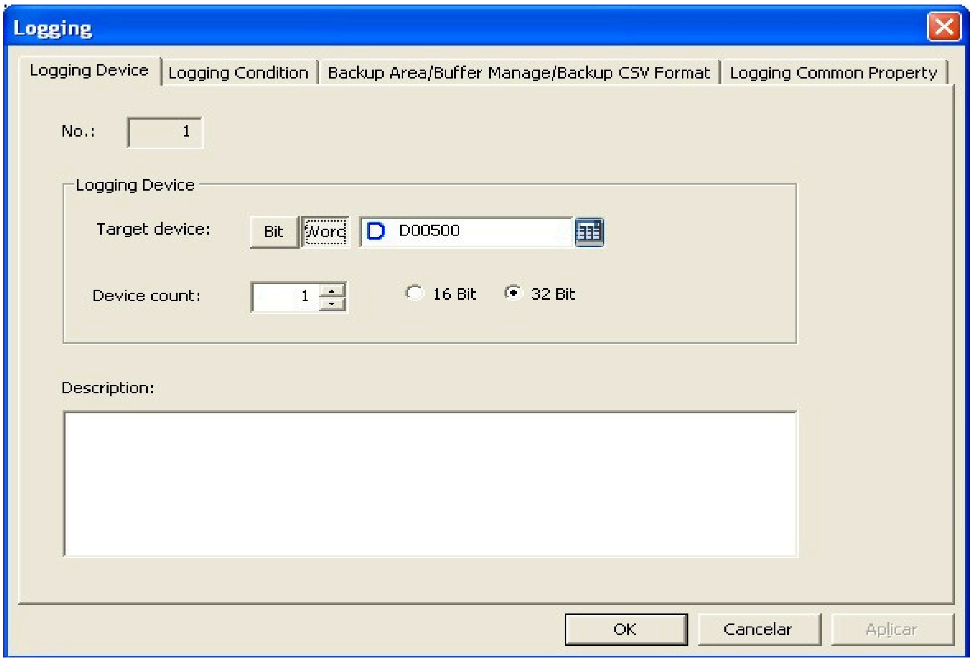


Now double click on blank line 1 of the log:



No	Condition	Repeat Count	Repeat Period	Device
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				

- On the first screen of the "Logging Device" logging, you must configure which memory contains the data you want to back up.



No.: 1

Logging Device

Target device: Word D D00500

Device count: 1 ○ 16 Bit ● 32 Bit

Description:

Logging device: you must choose whether you want to back up a bit or a word. You also choose which memory will be copied; in this example, we chose D500 (PLC memory) and the word option.

If Bit is selected, the memory state will be backed up, whether it is at a high or low level. If Word is selected, the data contained in this word will be backed up.

Device count: In this option you must enter how many records you want to have in your spreadsheet.

The logging configures a sequence of two memories at a time, for example:

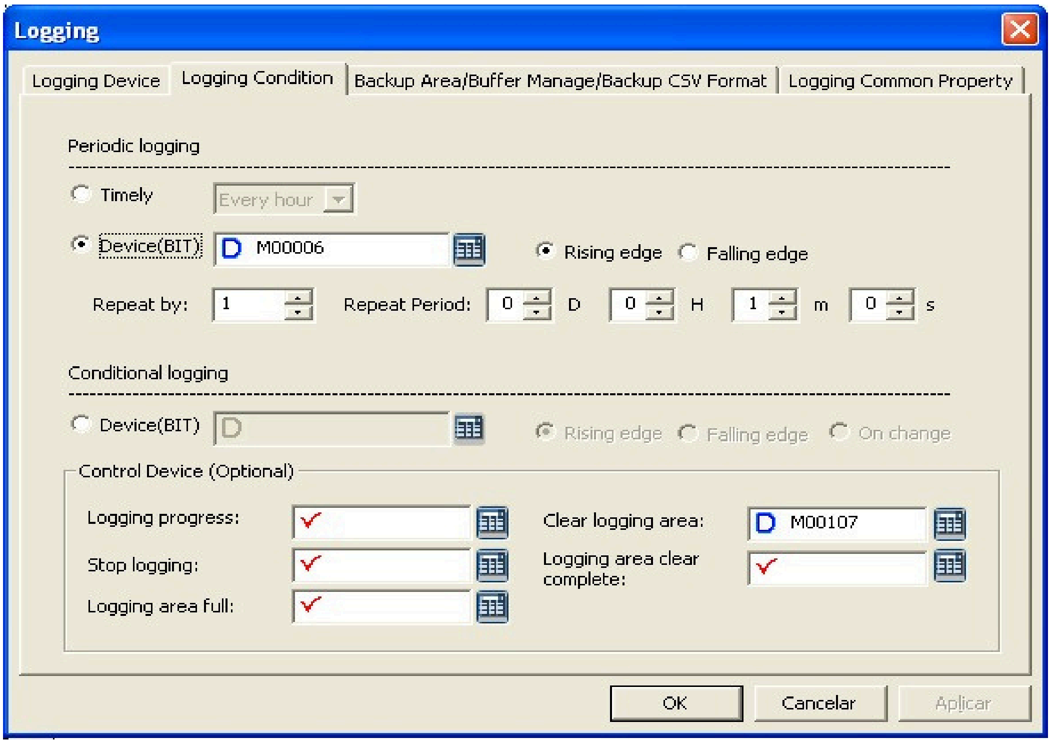
If you configured the Device count to 3, it will reserve six memories for backup.

Since in this example above the memory D500 was configured, if we had left

the Device count at 3, the logging would backup the data that is stored in memories

D500, D502 and D504.

- **In the next tab we have the conditions for the backup to happen.**



Periodic logging

○ Timely Every hour

● Device(BIT) M00006 ● Rising edge ○ Falling edge

Repeat by: 1 Repeat Period: 0 D 0 H 1 m 0 s

Conditional logging

○ Device(BIT) D ● Rising edge ○ Falling edge ○ On change

Control Device (Optional)

Logging progress:

Stop logging:

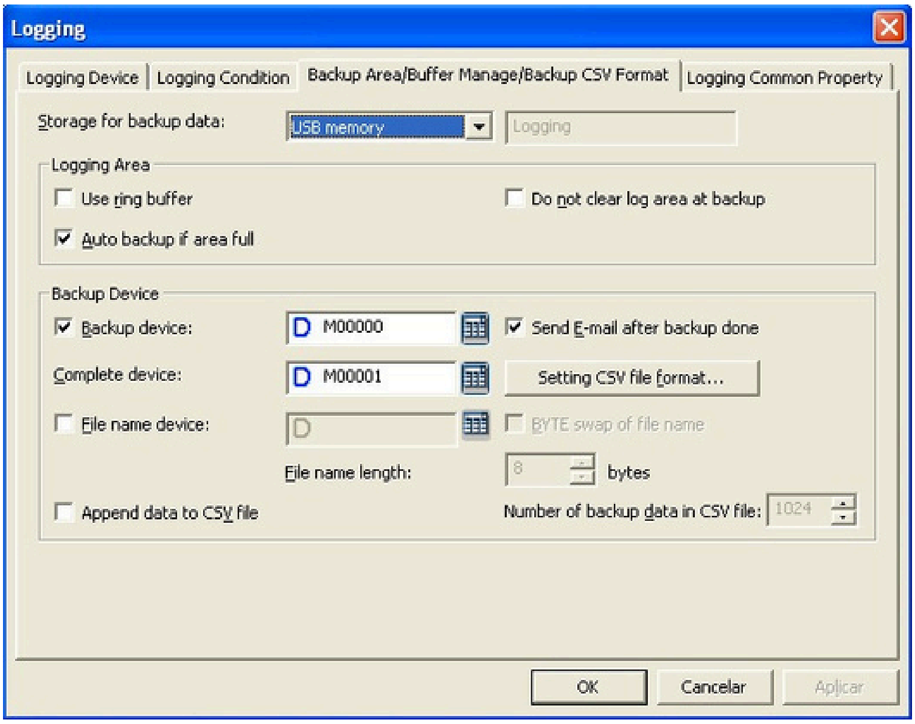
Logging area full:

Clear logging area: M00107

Logging area clear complete:

In Periodic logging you can configure how often the backup will be made or you can configure a memory to activate the backup initialization. In this example we have configured memory M6, which means that every time memory M6 goes to a high level (from 0 to 1) the data contained in memory D500 will be copied and written to the HMI's internal memory. The options Rising edge and Falling edge, when checked, mean that the backup will be made on the rising edge of memory M6 or the backup will be made on the falling edge of M6, respectively. With Repeat by and Repeat Period, you configure how many times in a given period of time the backup will be allowed. For example, if Repeat by is set to 1 and Repeat period in the minutes field is set to 1, as in the example above, the backup will be made once per minute, even if the M6 memory goes to a high logic level several times within this minute. This means that only 1 backup will be allowed during the current minute. The next acquisition will only be released in the next minute and will only be made when M6 goes to a high level again. Remember that it recognizes the rising edge, so M6 needs to go from logic level 0 to logic level 1 if the Rising edge option is checked.

In the Backup Area/Buffer Manage/Backup Format tab we can choose where the HMI will send the backup data stored in its internal memory and we can also configure how the data will appear in the spreadsheet.



Storage for backup data: USB memory Logging

Logging Area

□ Use ring buffer

□ Do not clear log area at backup

☑ Auto backup if area full

Backup Device

☑ Backup device: D M00000

☑ Send E-mail after backup done

Complete device: D M00001

Setting CSV file format...

□ File name device: D

□ BYTE swap of file name

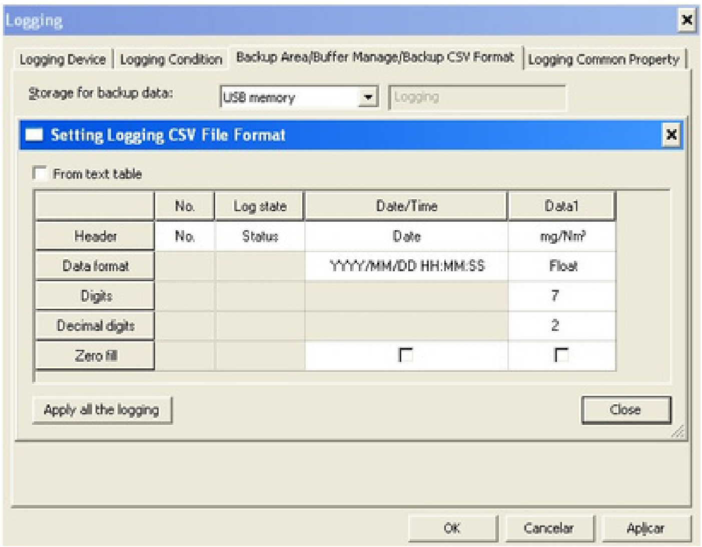
File name length: 8 bytes

□ Append data to CSV file

Number of backup data in CSV file: 1024

In this example, we selected the USB Memory option, which will send the backup information that is in the HMI's internal memory to a USB flash drive. However, in order for this information to be sent to the USB flash drive, we need to select the Backup device option and configure a memory (bit) in the field next to it. This means that when this memory goes to a high level, the HMI starts the transfer process to the external device. In Logging Area, we have two important options. The HMI is configured at the factory so that after transferring the data that is in its internal memory to an external device, the system automatically clears its internal memory for the next acquisitions. Now, if the Do not clear log area at backup option is selected, this will not occur and when you transfer a copy of the file containing the collected data to the external device, the original file will remain taking up space in the HMI's internal memory.

If the HMI's internal memory is completely occupied and the Auto backup If area full option is checked, the HMI system will automatically make a backup to the configured external device. If these two options are not checked and for some reason an error occurs when sending data to the external device, this data will still be stored in the HMI's internal memory, even if it is turned off, and can be sent to an external device at any time. In Backup Device, in the example above, we configured memory M0, which means that when M0 goes to a high logic level, all data in the HMI's internal memory will begin to be transferred to the flash drive. After this data is copied, the HMI notifies the end of the copy by activating a bit. In this case, we configured memory M1, where this memory can be an LED on the HMI screen, for example, to show that the transfer process has been completed. The Send Email after backup done option can be checked if you want the file with the backup data to be sent by email. Remember that for this we also need to configure other fields, which will be shown at the end of this procedure. ⍰ Still in Backup Device, we have a button called Settings CSV file format. Clicking on this button, we have the following screen:

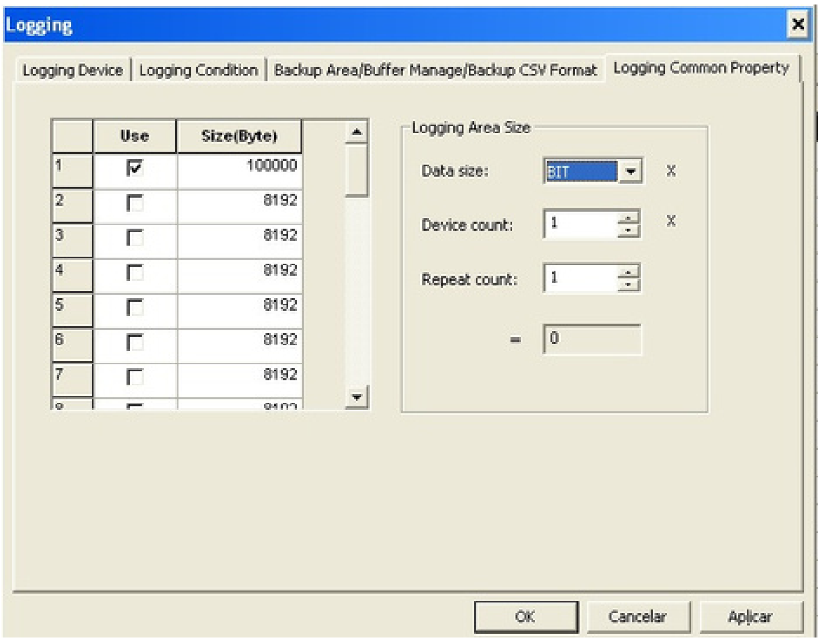


Storage for backup data: USB memory Logging

□ From text table

	No.	Log state	Date/Time	Data1
Header	No.	Status	Date	mg/Nm³
Data format			YYYY/MM/DD HH:MM:SS	Float
Digits				7
Decimal digits				2
Zero fill			□	□

On this screen, we can configure how the data will appear on the spreadsheet. In the Data 1 column, the first line will be the name of the column that will contain the collected data. In this example, we call it mg/Nm³. The lines below are configured, respectively: the type of data you are collecting, in this case float, with a maximum of 7 digits and always showing 2 decimal places after the comma. If we had configured the Device count, on the first logging screen, other than 1, more configurable columns like this would be shown on this screen, named as Data 1, Data 2, Data 3, etc... and so on.



	Use	Size(Byte)
1	☑	100000
2	□	8192
3	□	8192
4	□	8192
5	□	8192
6	□	8192
7	□	8192

Logging Area Size

Data size: BIT X

Device count: 1 X

Repeat count: 1

= 0

In the last tab we have to configure the internal memory space that we need to reserve in the HMI. The maximum configurable memory is described in the manual of each HMI model. Remember that when we start collecting data, the HMI first sends this data to an internal memory, in this case we reserve approximately 100 Kbytes of internal memory.

After these configurations, just click Ok and test the application.

Configuration for sending email

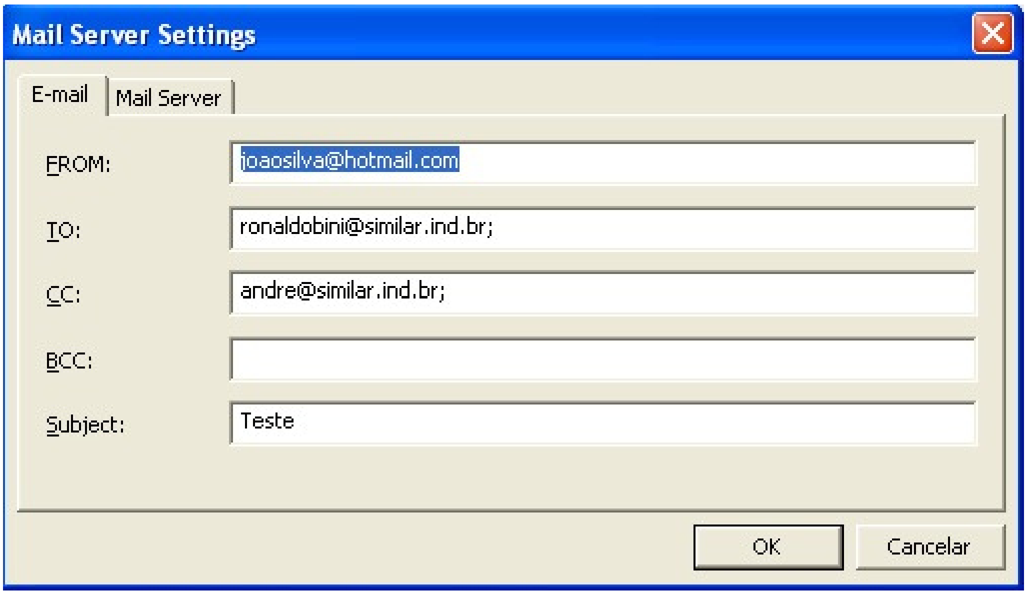
- Still in XP-Builder, click on Common > Project Property Settings;

- Now click on the Auxiliary Settings tab and check the Use E-mail

function option;

- Click on the Server Settings button;

- Configure the e-mail data as shown in the screen below:



FROM: joaosilva@hotmail.com

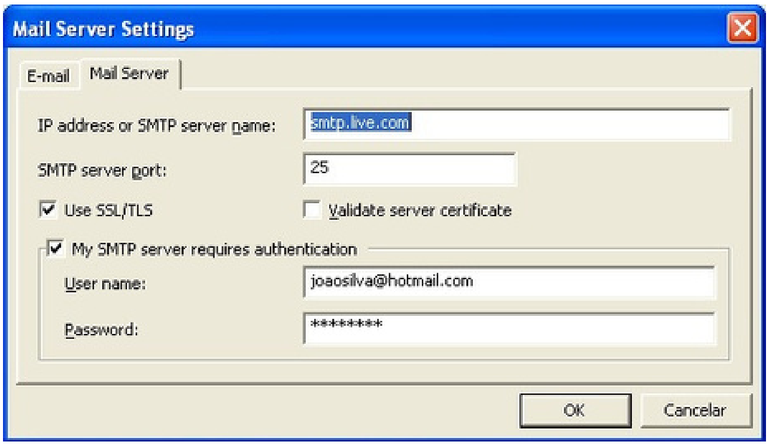
TO: ronaldobini@similar.ind.br;

CC: andre@similar.ind.br;

BCC:

Subject: Teste

- In the Mail Server tab, you must configure the SMTP sending server of the email account used and the sending port.



IP address or SMTP server name: smtp.live.com

SMTP server port: 25

☑ Use SSL/TLS ☐ Validate server certificate

☑ My SMTP server requires authentication

User name: joaosilva@hotmail.com

Password: ********

In the case of Hotmail, we need to select the Use SSL/TLS option because the Hotmail server requires this type of authentication, but this will depend on the email server used. We also need to select the My SMTP option and configure the username and password of the email that will send the attached file. After these settings, just click OK and every time the system backs up the HMI's internal memory to an external device, an email will be automatically sent with an attached file of the collected data.

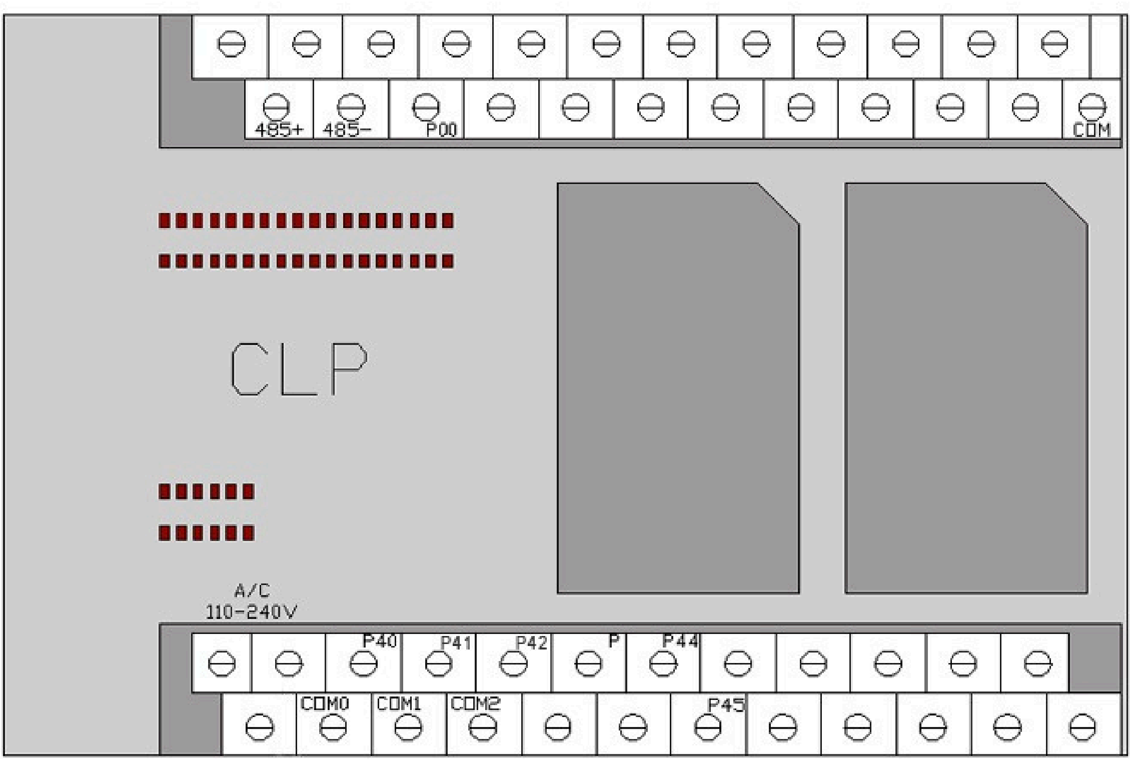
And.

□ Positioning - Stepper Motor with PLC

The PLC used for this type of application must necessarily have transistor outputs.

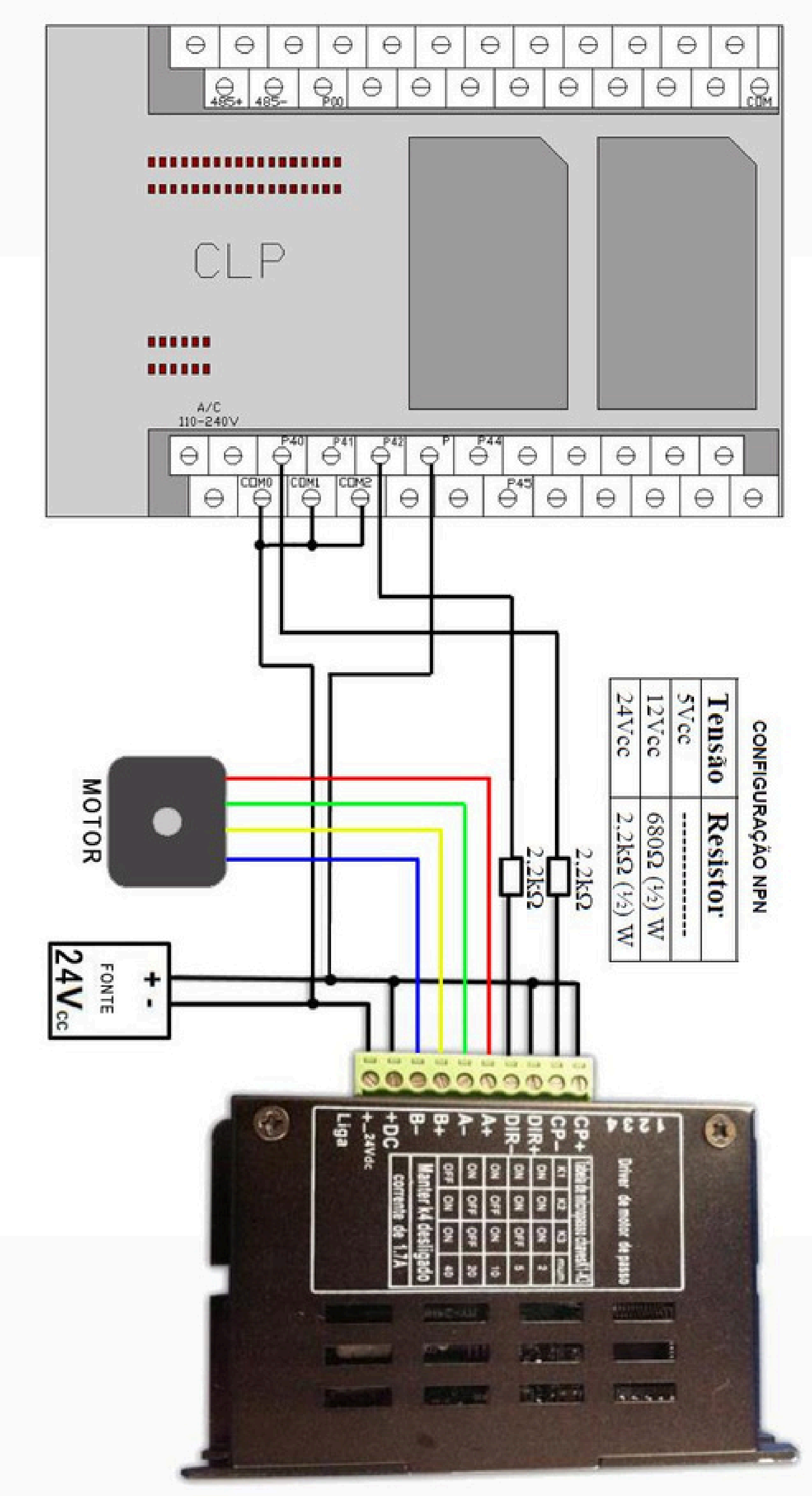
WIRING DIAGRAM:

We need to know whether the output of the PLC used is NPN or PNP. In the case of LS PLCs, all outputs are NPN.



In this PLC, output P40 is responsible for the "Pulses" of the X-axis, and output P42 is responsible for the "Direction" of the X-axis. For the Y-axis, outputs P41 are used for "Pulse" and P43 for "Direction".

Connection diagram: PLC – Driver – Motor:



CONFIGURAÇÃO NPN

Tensão	Resistor
5Vcc	---------------
12Vcc	680Ω (½) W
24Vcc	2,2kΩ (½) W

tabela de micropasso chaves(K1-K3)

K1	K2	K3	num
ON	ON	ON	2
ON	ON	OFF	5
ON	OFF	ON	10
ON	OFF	OFF	20
OFF	ON	ON	40

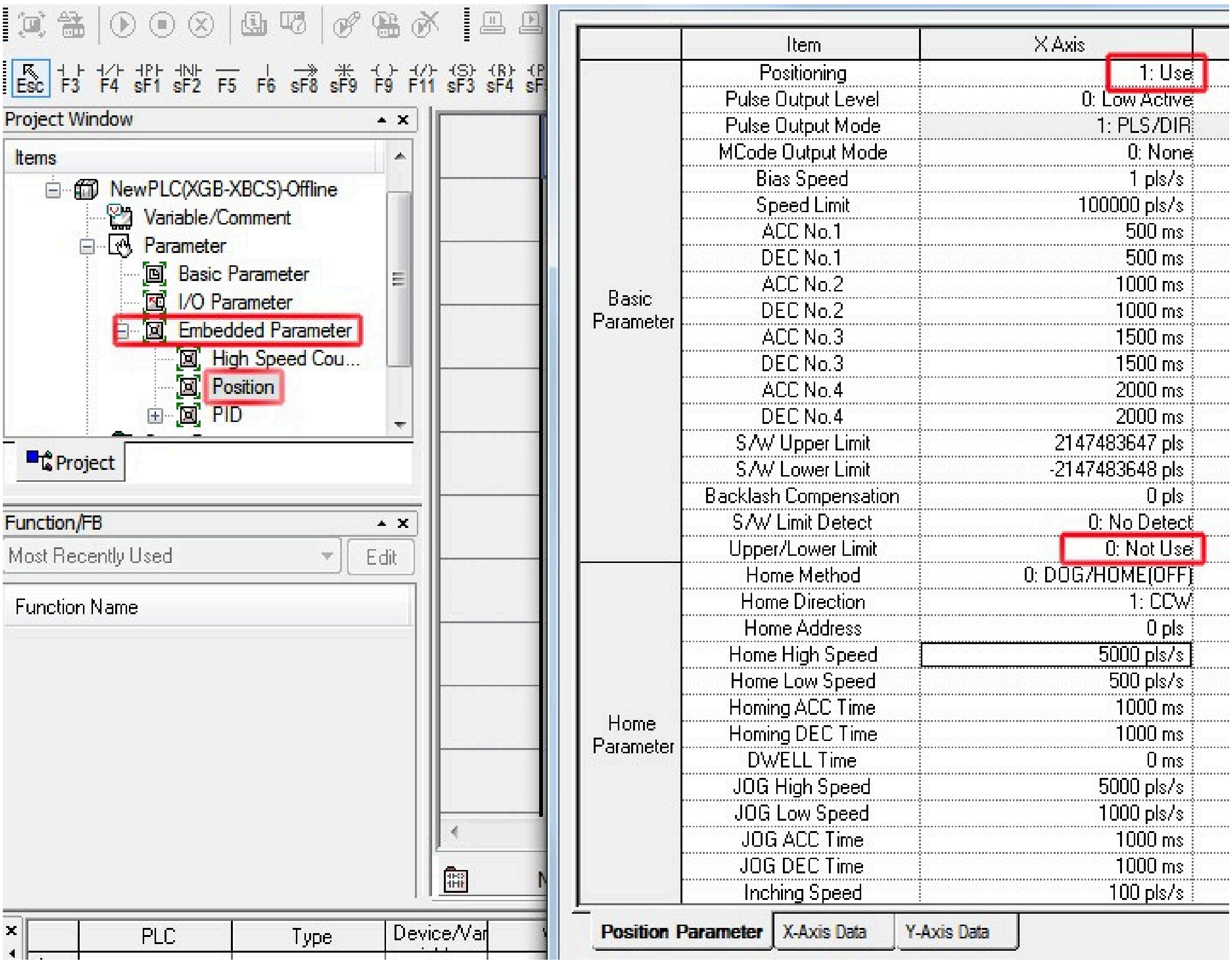
Manter k4 desligado

corrente de 1.7A

Resistors must be used at the CP and DIR inputs of the driver to reduce the voltage, as this drive works with 5V at its inputs and the PLC in this case supplies 24V at its outputs.

Configuration in XG5000:

Click on "Embedded Parameter" > "Position"

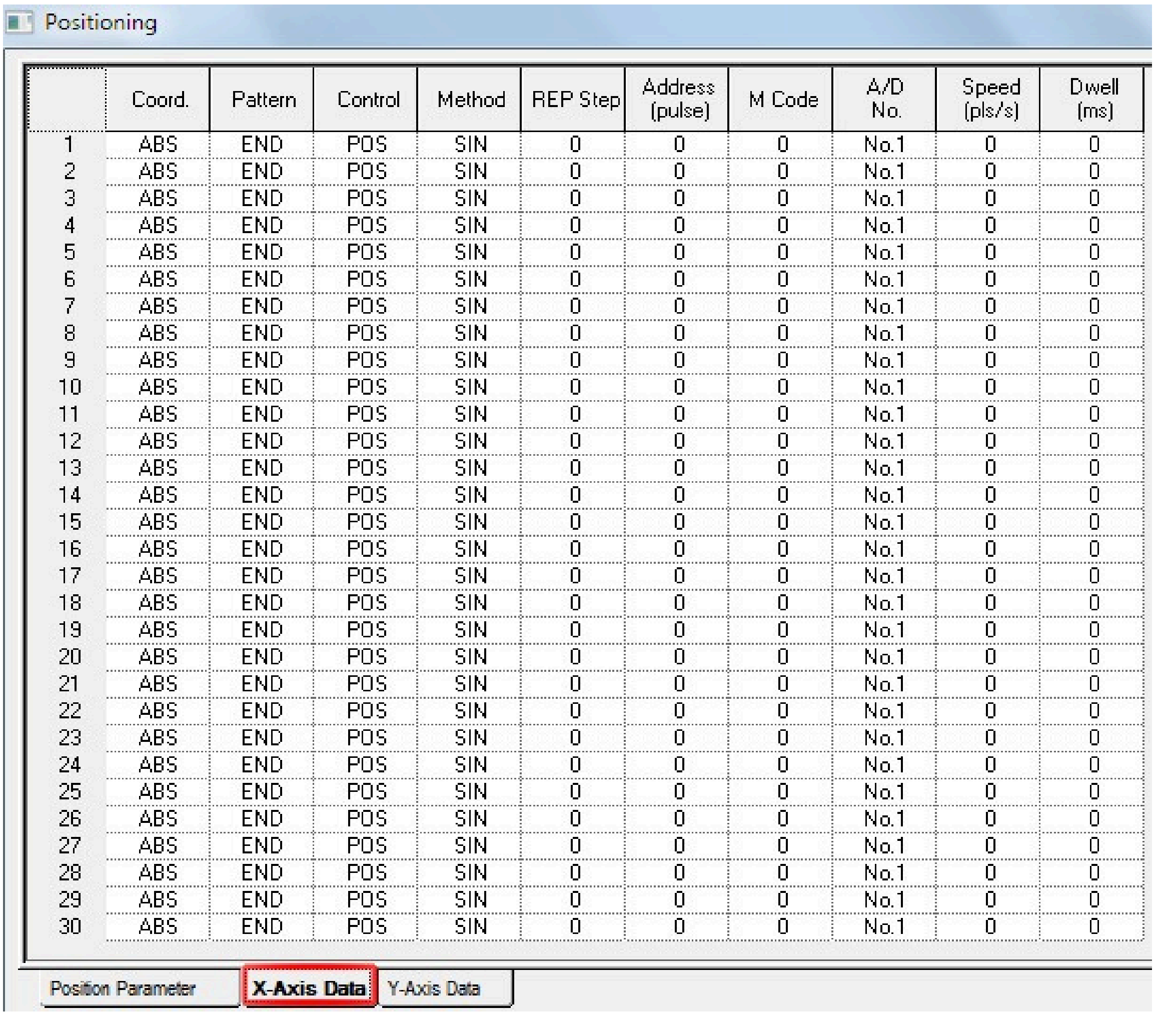


	Item	X Axis
Basic Parameter	Positioning	1: Use
	Pulse Output Level	0: Low Active
	Pulse Output Mode	1: PLS/DIR
	MCode Output Mode	0: None
	Bias Speed	1 pls/s
	Speed Limit	100000 pls/s
	ACC No.1	500 ms
	DEC No.1	500 ms
	ACC No.2	1000 ms
	DEC No.2	1000 ms
	ACC No.3	1500 ms
	DEC No.3	1500 ms
	ACC No.4	2000 ms
	DEC No.4	2000 ms
	S/W Upper Limit	2147483647 pls
	S/W Lower Limit	-2147483648 pls
	Backlash Compensation	0 pls
	S/W Limit Detect	0: No Detect
	Upper/Lower Limit	0: Not Use
Home Parameter	Home Method	0: DOG/HOME(OFF)
	Home Direction	1: CCW
	Home Address	0 pls
	Home High Speed	5000 pls/s
	Home Low Speed	500 pls/s
	Homing ACC Time	1000 ms
	Homing DEC Time	1000 ms
	DWELL Time	0 ms
	JOG High Speed	5000 pls/s
	JOG Low Speed	1000 pls/s
	JOG ACC Time	1000 ms
	JOG DEC Time	1000 ms
	Inching Speed	100 pls/s

In this example, we will enable only the X-axis to work with the IST and DST functions, which will be explained later. In "Positioning", change to "1: Use" to enable the X-axis, and in "Upper/Lower Limit", change to "0: Not Use" because in this example, we will not use maximum and minimum limit sensors.

There are two ways to control the Stepper Motor with LS PLCs. The "Position" function can be controlled either with the IST (Indirectly Start) function or with the DST (Directly Start) function.

PLC PROGRAMMING IN IST MODE:



	Coord.	Pattern	Control	Method	REP Step	Address (pulse)	M Code	A/D No.	Speed (pls/s)	Dwell (ms)
1	ABS	END	POS	SIN	0	0	0	No.1	0	0
2	ABS	END	POS	SIN	0	0	0	No.1	0	0
3	ABS	END	POS	SIN	0	0	0	No.1	0	0
4	ABS	END	POS	SIN	0	0	0	No.1	0	0
5	ABS	END	POS	SIN	0	0	0	No.1	0	0
6	ABS	END	POS	SIN	0	0	0	No.1	0	0
7	ABS	END	POS	SIN	0	0	0	No.1	0	0
8	ABS	END	POS	SIN	0	0	0	No.1	0	0
9	ABS	END	POS	SIN	0	0	0	No.1	0	0
10	ABS	END	POS	SIN	0	0	0	No.1	0	0
11	ABS	END	POS	SIN	0	0	0	No.1	0	0
12	ABS	END	POS	SIN	0	0	0	No.1	0	0
13	ABS	END	POS	SIN	0	0	0	No.1	0	0
14	ABS	END	POS	SIN	0	0	0	No.1	0	0
15	ABS	END	POS	SIN	0	0	0	No.1	0	0
16	ABS	END	POS	SIN	0	0	0	No.1	0	0
17	ABS	END	POS	SIN	0	0	0	No.1	0	0
18	ABS	END	POS	SIN	0	0	0	No.1	0	0
19	ABS	END	POS	SIN	0	0	0	No.1	0	0
20	ABS	END	POS	SIN	0	0	0	No.1	0	0
21	ABS	END	POS	SIN	0	0	0	No.1	0	0
22	ABS	END	POS	SIN	0	0	0	No.1	0	0
23	ABS	END	POS	SIN	0	0	0	No.1	0	0
24	ABS	END	POS	SIN	0	0	0	No.1	0	0
25	ABS	END	POS	SIN	0	0	0	No.1	0	0
26	ABS	END	POS	SIN	0	0	0	No.1	0	0
27	ABS	END	POS	SIN	0	0	0	No.1	0	0
28	ABS	END	POS	SIN	0	0	0	No.1	0	0
29	ABS	END	POS	SIN	0	0	0	No.1	0	0
30	ABS	END	POS	SIN	0	0	0	No.1	0	0

In this mode, we have a table that can be configured to execute a sequence of steps. It is commonly used when the stepper motor will always perform the same movements in a pre-defined sequence.

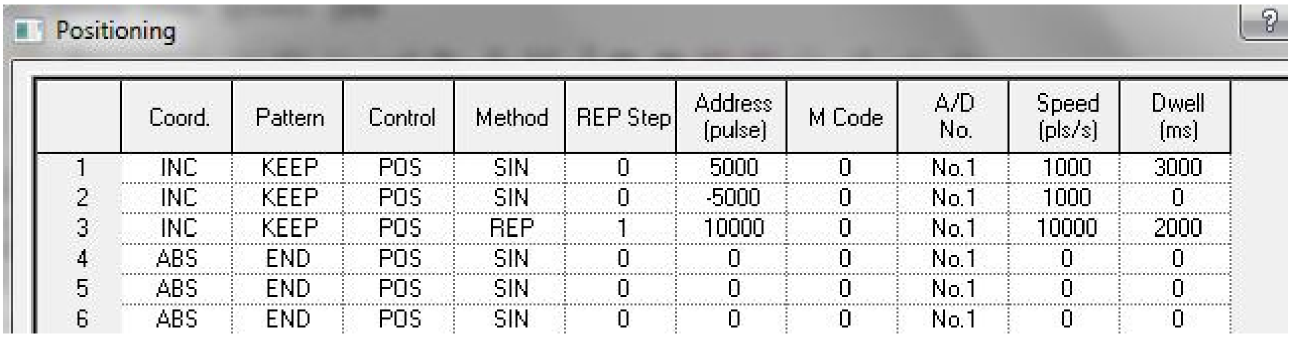
1st Column - Step - In the first column, we find the step number, which represents the steps that will be executed according to the step configuration. The number of steps may vary depending on the PLC used. In this example, we can see that we can perform a task with up to 30 movements on the X-axis and 30 movements on the Y-axis when using the IST function.

2nd Column – Coord. - In this option, the type of coordinate is configured: Absolute (ABS) or Incremental (INC).

3rd Column – Pattern - In this function, there are two commonly used options:

- **END: Executes the step in this line and stops; it does not proceed to the next one.**
- **KEEP: Executes the step in this line and continues to the next line in sequence, and so on, until it encounters a line set to END.**
- **4th Column – Control - In this column, you define whether the control for this step will be based on position or speed.**
- **5th Column – Method - This column is used when you want to repeat the programming starting from a specific step.**
- **Example: If the sequence of movements in your program ends at step 6, but you need to restart the program and go back to the first step, making it repeat the sequence of steps 1, 2, 3, 4, 5, and 6 again. In step 6, you can configure the Method as "REP" and in the next column, configure the step number where you want the program to go (in our case, step number 1).**
- **6th Column – REP Step - Continuing the example above. This column specifies the step number where you want the program to go. In our example, we want the program to perform a sequence of 6 movements, and once the sixth movement is completed, it should restart this sequence from the first step. Therefore, in this case, in the sixth step, we should place the number 1 in this column, indicating the start (step 1) and end (step 6) of a loop.**
- **7th Column – Address (pulse) - In this column, you need to define the number of pulses this line will execute. The stepper motor will move according to the number of pulses.**
- **Remember that to determine the distance traveled, it depends not only on the number of pulses but also on how many degrees the stepper motor rotates with just one pulse and the diameter of the shaft.**
- **8th Column – M code - Always leave it at 0.**
- **9th Column – A/D No. - This column is responsible for the acceleration and deceleration ramp, which is configured in the Position Parameter tab. You can configure up to 4 ramps.**
- **10th Column – Speed (pls/s) - This column is responsible for the pulse speed (pulses per second) sent by the PLC. In this tutorial, we are using output P40. The more pulses per second the PLC sends to the stepper motor driver, the faster the motor will rotate, respecting the limits of both the driver and the PLC.**

11th Column – Dwell (ms) – In this column you can set a delay time before starting to execute the next line. For example: If the first line is set to Dwell of 500 milliseconds, the program will execute the first line, wait half a second and then execute the second line. This is a time delay between lines. Below is an example of 3 movements made in the IST function:



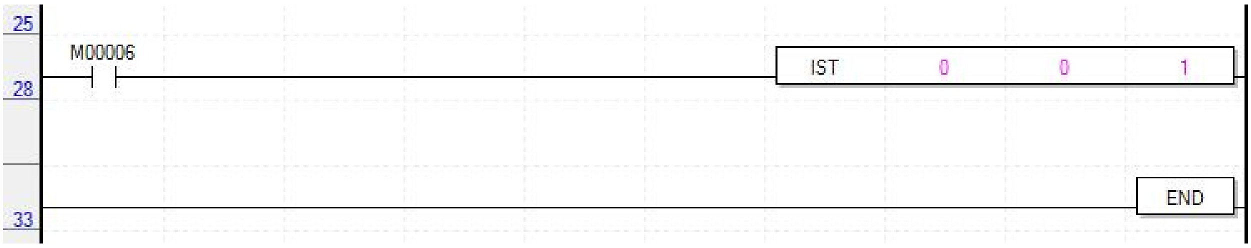
	Coord.	Pattern	Control	Method	REP Step	Address (pulse)	M Code	A/D No.	Speed (pls/s)	Dwell (ms)
1	INC	KEEP	POS	SIN	0	5000	0	No.1	1000	3000
2	INC	KEEP	POS	SIN	0	-5000	0	No.1	1000	0
3	INC	KEEP	POS	REP	1	10000	0	No.1	10000	2000
4	ABS	END	POS	SIN	0	0	0	No.1	0	0
5	ABS	END	POS	SIN	0	0	0	No.1	0	0
6	ABS	END	POS	SIN	0	0	0	No.1	0	0

Step 1: The movement will start in incremental coordinates, moving the stepper motor 5000 pulses at a speed of 1000 pulses per second. After completing this distance, it will wait for 3 seconds before executing Step 2.

Step 2: The movement will continue, but now in the opposite direction, as it will go to address -5000. Since the motor is currently at pulse 5000, it will move back -5000 pulses, returning to the origin position (0) and without waiting any time, it will proceed to Step 3.

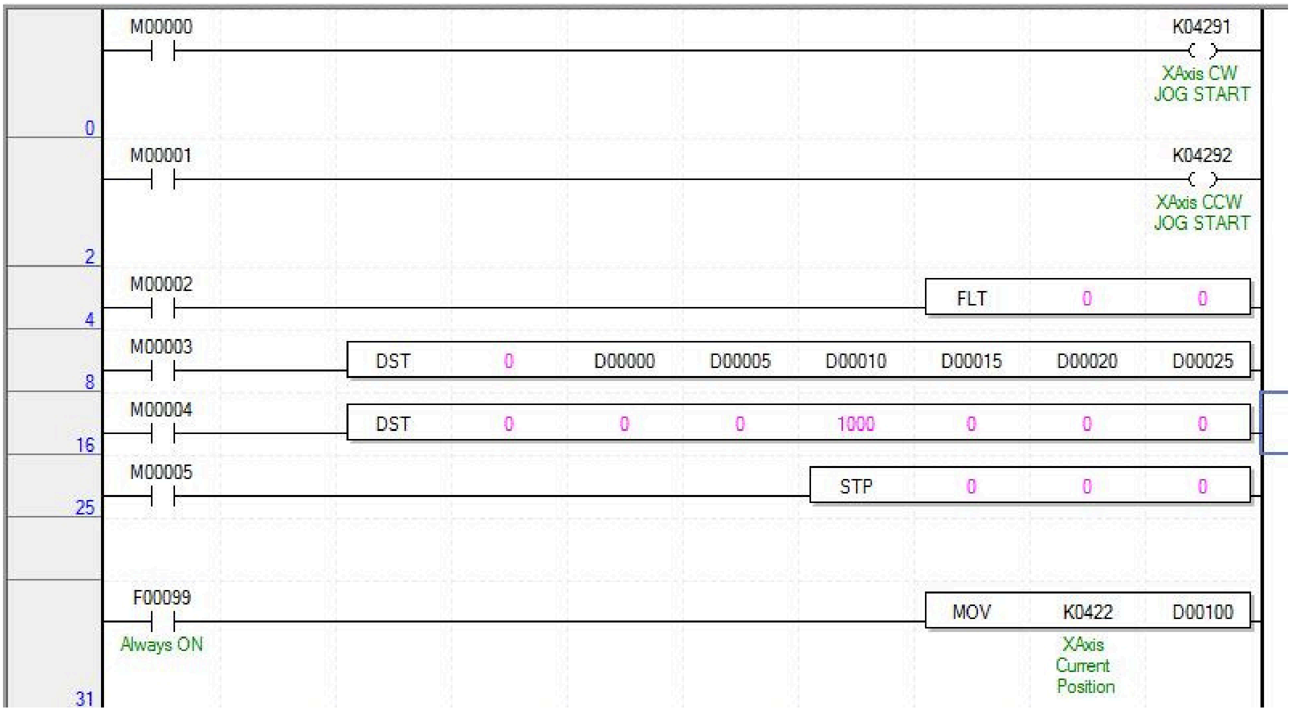
Step 3: In the third line, the motor will move 10000 pulses in the clockwise direction at a speed of 10000 pulses per second, wait for 2 seconds, and then go to Step 1, repeating all movements in a loop.

To activate the IST function, we place an M6 memory, where the IST function, like the other Positioning functions, will only recognize the rising edge of this contact, so we can use the F3 or sF1(P) contact of the XG5000.



Depois da tabela configurada, fica muito simples utilizar a função IST. Para isso bastar apertar a tecla F10 do teclado e escrever IST. Descrição dos parâmetros : 1° Parâmetro (0) - sl – O primeiro parâmetro é responsável pelo número do SLOT. O slot 0 significa que o driver do motor de passo está ligado direto no CLP. Caso seja utilizado um módulo de expansão de posicionamento conectado ao CLP, o número do slot será o número referente a quantidade de módulos que se encontram conectados no CLP, por exemplo: Se você possuí um CLP e três módulos conectado a sua lateral e o módulo de posicionamento é o terceiro módulo, o número do slot será 3 e assim por diante. 2° Parâmetro (0) - ax – Este parâmetro é responsável pelo eixo que você deseja movimentar, para o eixo X você deve entrar com o valor "0", caso queria comandar o eixo Y o valor do segundo parâmetro deve ser "1". 3° Parâmetro: (1) (n1) – Este parâmetro é responsável pelo número do step a ser executado. O número 1 indica que o programa vai iniciar a execução pela primeira linha. Ao acionarmos a memória M6 iniciará o movimento da primeira linha e caso esta primeira linha estiver configurada como KEEP ela irá terminar o movimento e seguir para a segunda linha e assim por diante. Lembrando que para o movimento do motor acontecer, precisamos antes dar um pulso na função FLT para mostrar ao programa onde é a origem do movimento e só depois desta ação o IST irá funcionar.

PLC PROGRAMMING IN DST MODE:



```
0   M00000                                                          K04291
    --| |----------------------------------------------------------( )--
                                                                   XAxis CW
                                                                   JOG START
2   M00001                                                          K04292
    --| |----------------------------------------------------------( )--
                                                                   XAxis CCW
                                                                   JOG START
4   M00002
    --| |-------------------------------------------[ FLT   0       0 ]
8   M00003
    --| |-----[ DST  0  D00000  D00005  D00010  D00015  D00020  D00025 ]
16  M00004
    --| |-----[ DST  0  0       0       1000    0       0       0      ]
25  M00005
    --| |------------------------------------[ STP   0      0       0 ]

31  F00099
    --| |-------------------------------------------[ MOV  K0422  D00100 ]
    Always ON                                              XAxis
                                                           Current
                                                           Position
```

K04291 – Has the JOG function in the clockwise direction, this function is normally used to make the motor rotate when pressing a Push-Button type button and stop rotating when releasing the button. In the HMI it can be used with a "Momentary" type button (Bit Switch):

K04292 – Has the same function as the one mentioned above, but for the counterclockwise direction.

FLT – The FLT function is used to reset the current position. When memory M2, in this example, goes to logic level 1, the program will understand that this is the origin of the path and will send the pulse number of the current position to zero, being ready to receive the next position (number of pulses) to move to, considering that it is starting from pulse zero. The FLT function must receive a pulse the first time the program is executed. It can also be used again if you want to reset the pulses of the current position.

DST – The first DST function we have in the program is being activated by memory M0003. This function aims to provide the program with all the parameters necessary for the stepper motor to start operating. You can enter fixed values or memories in this function so that the user can later enter the values via the HMI. In this example, we did both methods to demonstrate. In this first DST function, we have: 1st Parameter: 0 (sl) – The first parameter is responsible for the SLOT number. Slot 0 means that the stepper motor driver is connected directly to the PLC. If a positioning expansion module is used and connected to the PLC, the slot number will be the number corresponding to the number of modules connected to the PLC. For example: If you have a PLC and three modules connected to its side and the positioning module is the third module, the slot number in the DST function will be 3, and so on. 2nd Parameter: D0000 (ax) – This parameter is responsible for the axis you want to move. For the X axis, you must enter the value "0". If you want to control the Y axis, the value of the second parameter must be "1". 3rd Parameter: D0005 (Target Position) – This parameter is used to configure the position. It is the number of pulses that the PLC will send to the stepper motor driver. The position will depend on how many degrees the stepper motor rotates with one pulse. 4th Parameter: D0010 (Target Speed) – This parameter is responsible for the speed of pulses per second sent by the PLC output. In this tutorial, the output we are using is P40. The more pulses per second the PLC sends to the stepper motor driver, the faster the motor will rotate. Respecting the limits of the stepper motor driver and the PLC. 5th Parameter: D0015 (Dwell Time) – This parameter provides a time delay after the operation is performed. It is most commonly used when using the table mentioned above (X-Axis Data), where you configure a number of lines to perform a series of sequences. Then, when the program finishes executing a line, it waits for a configured time, in milliseconds, in the Dwell Time and only then starts the next line. If you leave this parameter at zero, the program will move on to the next step without a time delay. 6th Parameter: D0020 (Mcode) – Always leave it at zero. 7th Parameter: D0025 (Control Word) – This parameter is responsible for defining the type of coordinate you will use, Incremental or absolute, and is also responsible for the type of control, Position or Speed, and we need to configure it through the 16 bits of a Word, where we will use only bit 0 and bit 4:

Bit 0	Position: 0	Speed: 1
Bit 4	Absolute: 0	Incremental: 1

Following the reasoning in the table above, we have:

Type Position/ Absolute	**Binário**	**Hexadecimal (h)**
Speed/ Incremental	0000000000000000	0
Position/ Incremental	0000000000010001	11
Absolute/ Speed	0000000000010000	10
	0000000000000001	1

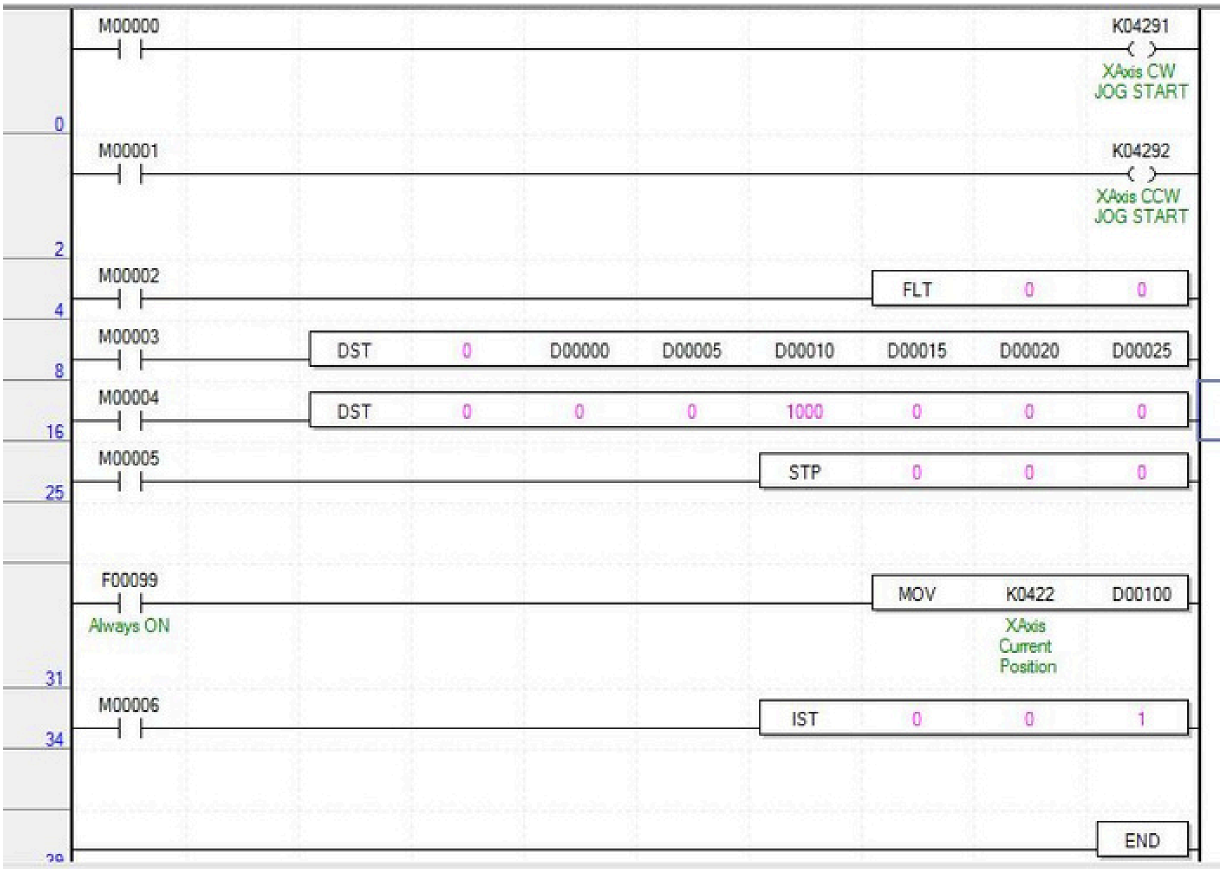
The number entered in this parameter must be in hexadecimal.

The next DST function that is being triggered by memory M0004 has the function of sending the motor back to the origin position (position 0). This origin position is the same position when the FLT function was previously triggered, indicating where position 0 would be.

STP – Stop function, used to stop the motor from rotating. The first parameter refers to the slot number, the second parameter to the axis (X = 0 / Y = 1) and the third parameter is the deceleration time.

K0422 – Shows the current position of the X axis.

Full Schedule:

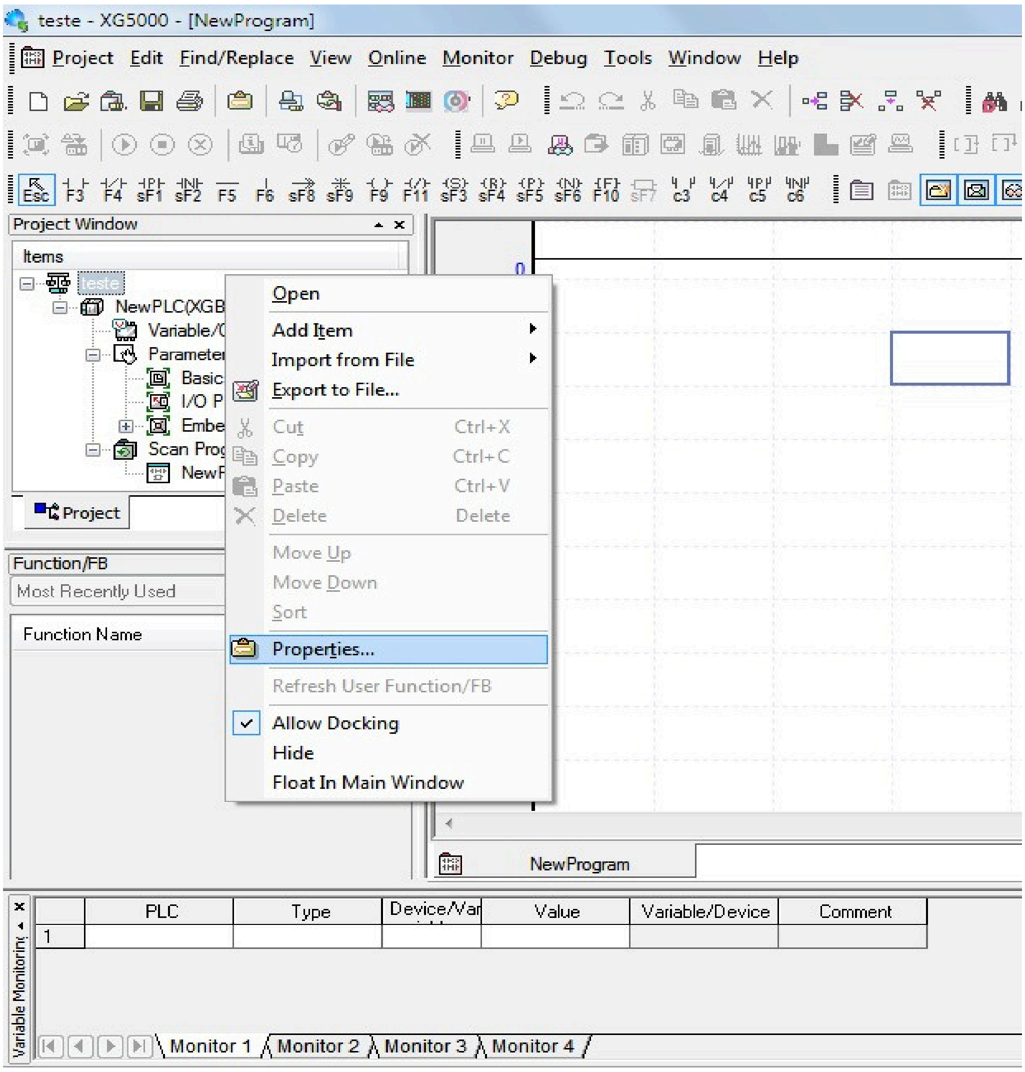


```
0   M00000                                                                  K04291
    -| |--------------------------------------------------------------------( )
                                                                         XAxis CW
                                                                         JOG START
2   M00001                                                                  K04292
    -| |--------------------------------------------------------------------( )
                                                                         XAxis CCW
                                                                         JOG START
4   M00002
    -| |--------------------------------------------------[FLT     0       0]
8   M00003
    -| |------[DST   0   D00000   D00005   D00010   D00015   D00020   D00025]
16  M00004
    -| |------[DST   0   0        0        1000     0        0        0     ]
25  M00005
    -| |-------------------------------------[STP      0        0        0]

31  F00099
    -| |--------------------------------------------------[MOV   K0422   D00100]
    Always ON                                                    XAxis
                                                                 Current
                                                                 Position
34  M00006
    -| |-------------------------------------[IST      0        0        1]

39                                                                       [END]
```

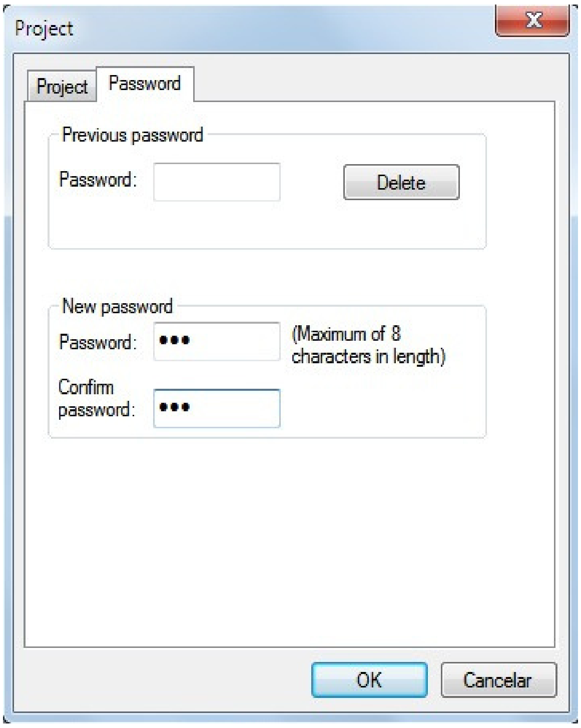
And

□- Password to “open file” on the XG5000: - On the XG5000: 1. Right-click on the project name and then click on properties:

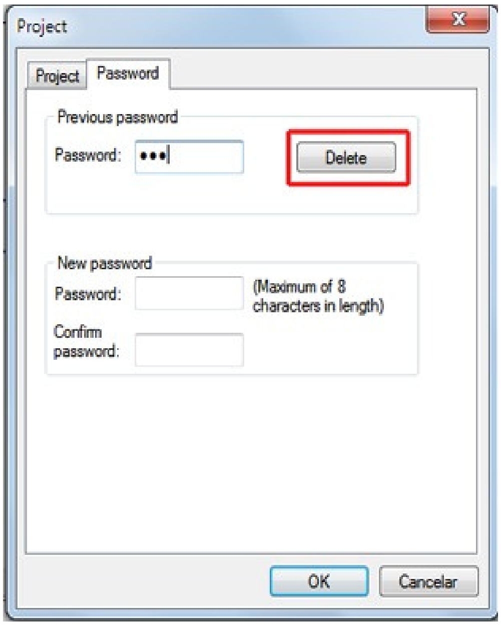
Password - XG5000 and PLC:



2. Enter the password, confirm and click OK:



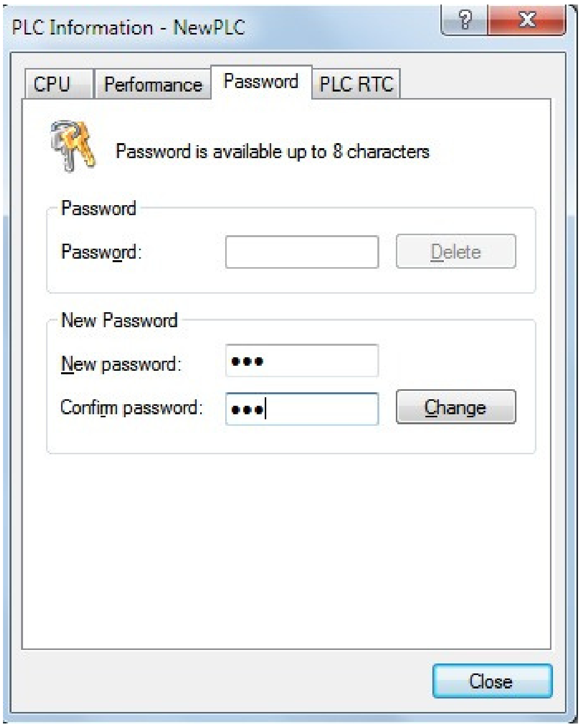
3. If the user needs to delete the password in the future, enter the password in the "Previous
password" field and click delete as shown in the figure below:



- Password for the “PLC Program”:

1- To set a password for PLC programming, after connecting to the PLC, on the XG5000, click on: Online > PLC Information > Password. Enter the password, confirm and click on Change.

–



Password is available up to 8 characters

Password:

New password: ●●●

Confirm password: ●●●

Done, the password has now been transferred to the PLC. If the user wants to connect to the PLC in the future

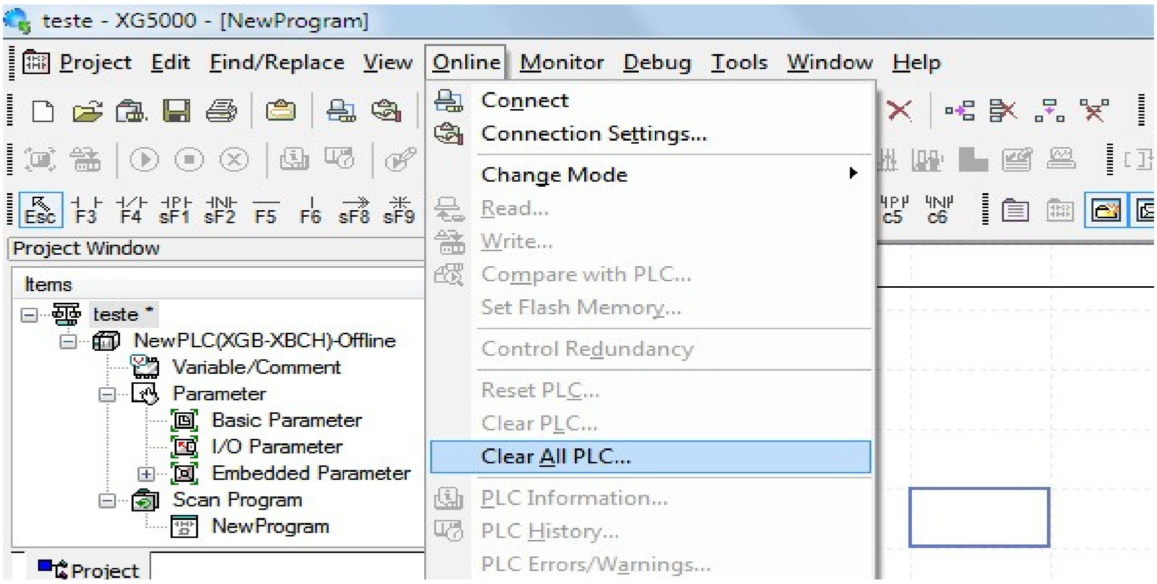
or wants to upload the program contained in it, the software will require the password.

To delete the password, follow step 3 on the previous page.

Clearing all the PLC parameters:

1- **If the user forgets the password that was set on the PLC, he/she has the option of clearing the PLC parameters. Just remember that this command deletes all passwords, parameters and any programs contained in it.**

Click on Online > Clear All PLC > Yes:



To perform this operation, the PLC does not need to be connected.

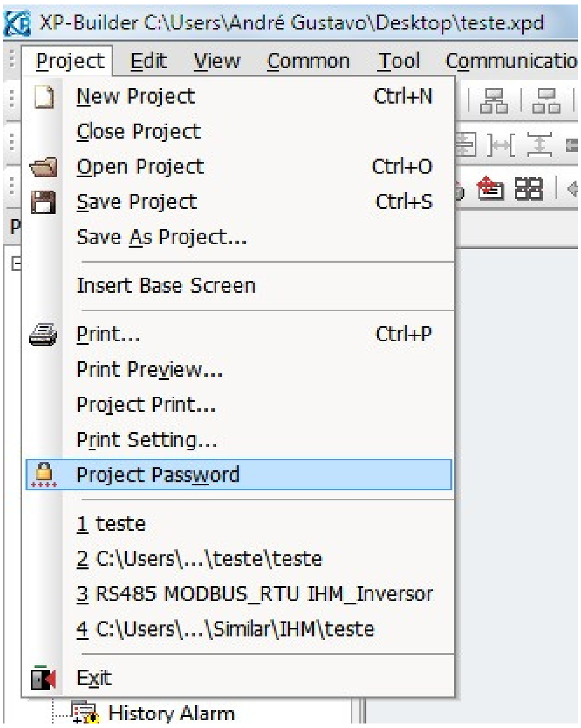
And

□

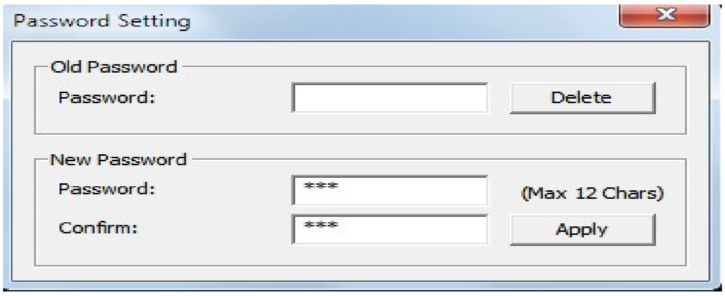
- In XP-Builder:

1. Click on Project > Project Password:

- Password for "opening the file or performing an upload of the HMI" in XP-Builder:
- Password - XP-Builder and HMI



2. Enter the password, confirm and click Apply:

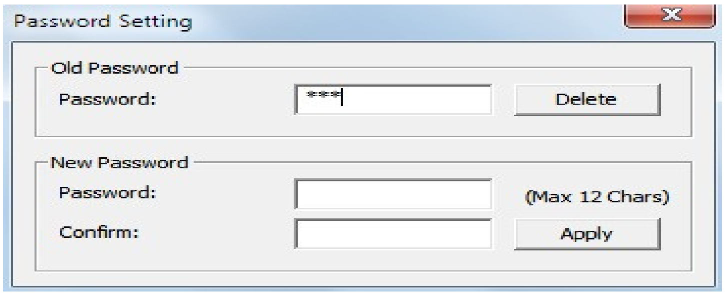


Password: ***

Confirm: ***

Transfer the program to the HMI. Now, to open or perform an upload of the program, the software will require the configured password.

3. If in the future the user needs to delete the password, enter the password in the "Old password" field and click on Delete, as shown in the figure below:



Password Setting

Old Password

Password: ***

New Password

Password:

(Max 12 Chars)

Confirm:

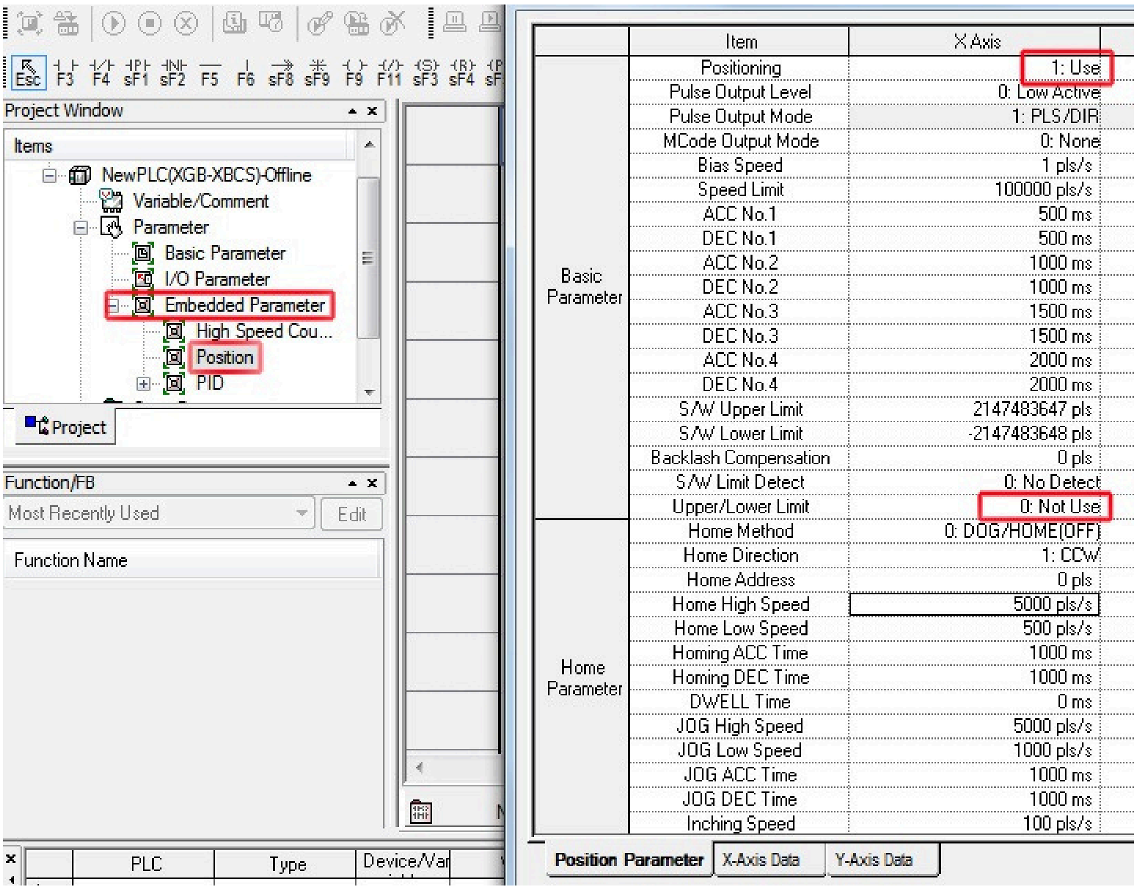
4. Transfer the program to the HMI.

And

Configuration in XG5000:

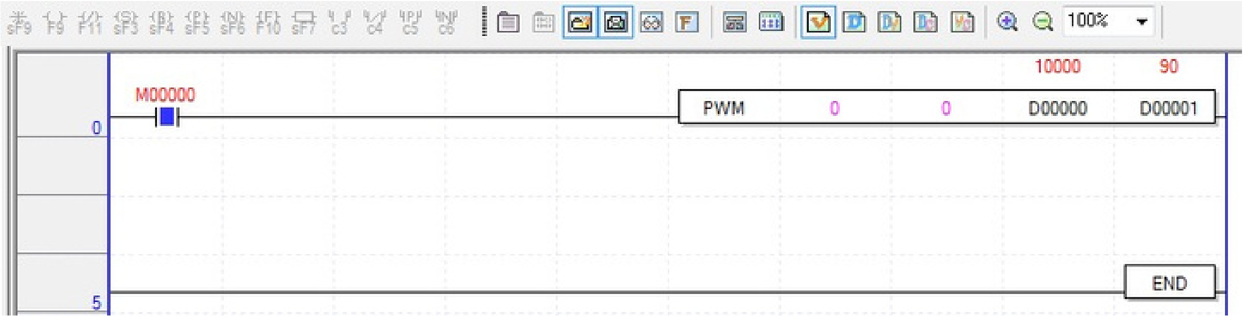
Click on “Embedded Parameter” > “Position”:

PWM:



	Item	X Axis
Basic Parameter	Positioning	1: Use
	Pulse Output Level	0: Low Active
	Pulse Output Mode	1: PLS/DIR
	MCode Output Mode	0: None
	Bias Speed	1 pls/s
	Speed Limit	100000 pls/s
	ACC No.1	500 ms
	DEC No.1	500 ms
	ACC No.2	1000 ms
	DEC No.2	1000 ms
	ACC No.3	1500 ms
	DEC No.3	1500 ms
	ACC No.4	2000 ms
	DEC No.4	2000 ms
	S/W Upper Limit	2147483647 pls
	S/W Lower Limit	-2147483648 pls
	Backlash Compensation	0 pls
	S/W Limit Detect	0: No Detect
	Upper/Lower Limit	0: Not Use
Home Parameter	Home Method	0: DOG/HOME(OFF)
	Home Direction	1: CCW
	Home Address	0 pls
	Home High Speed	5000 pls/s
	Home Low Speed	500 pls/s
	Homing ACC Time	1000 ms
	Homing DEC Time	1000 ms
	DWELL Time	0 ms
	JOG High Speed	5000 pls/s
	JOG Low Speed	1000 pls/s
	JOG ACC Time	1000 ms
	JOG DEC Time	1000 ms
	Inching Speed	100 pls/s

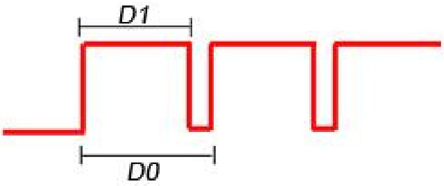
In this example, we will enable only the X-axis, but if you wish to work with the second fast output, enable the Y-axis in the same way.



M00000 – Activates the PWM function.

D00000 – Total cycle period in milliseconds.

D00001 – Percentage of the "total cycle" in which the signal will be at a high logic level.



In this example, we can conclude that the total cycle time is 10 seconds, and the output will remain high logic level for 9 seconds and low logic level for 1 second.

Note: Whenever the time (D00000) or the percentage of the high logic level (D00001) is changed, the program will require that memory M00000 be deactivated and reactivated to update the change.

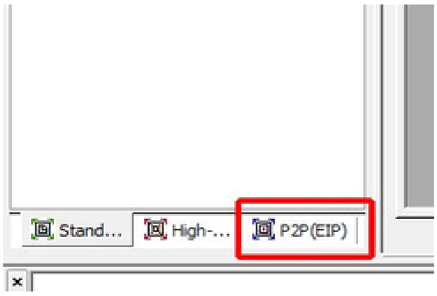
END.

□

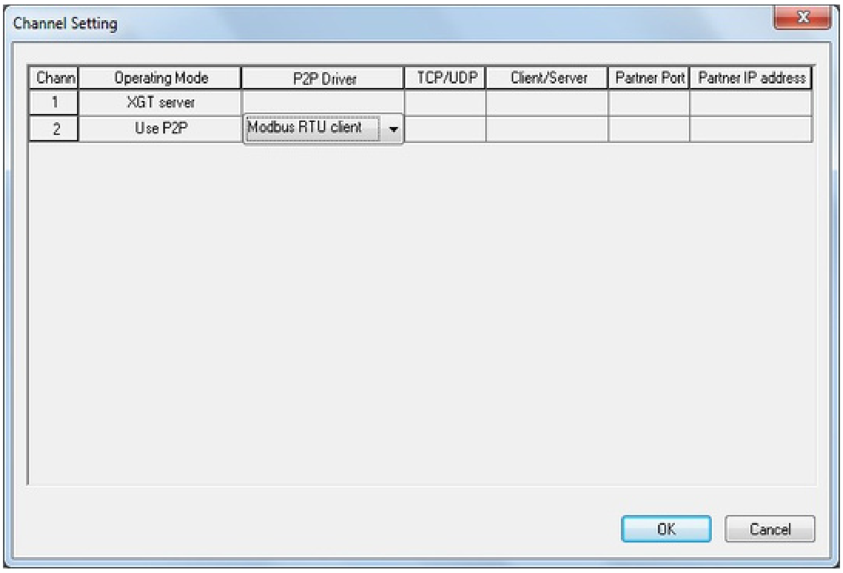
Reading Parameters on the Frequency Inverter:

- In XG5000:

1. Click on Tools > Network Manager;
2. New > Choose a name > Select the PLC model;
3. Double-click on 00: Embedded Cnet;
4. Configure Channel 2x according to the RS485 serial data (default baud rate: 9600) and in Channel 2, select: Use P2P;

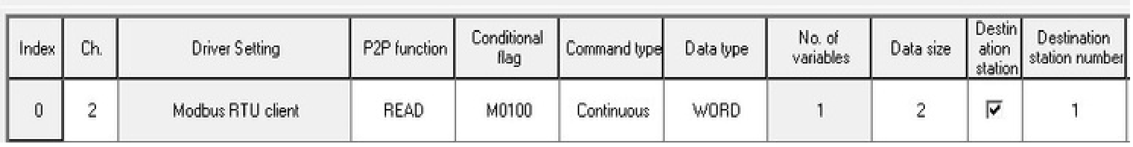


6 – Click twice on P2P 01 > OK; 7- Click twice on P2P Channel > In 2 Use > P2P Driver > Select: Modbus RTU client:



Chann	Operating Mode	P2P Driver	TCP/UDP	Client/Server	Partner Port	Partner IP address
1	XGT server					
2	Use P2P	Modbus RTU client				

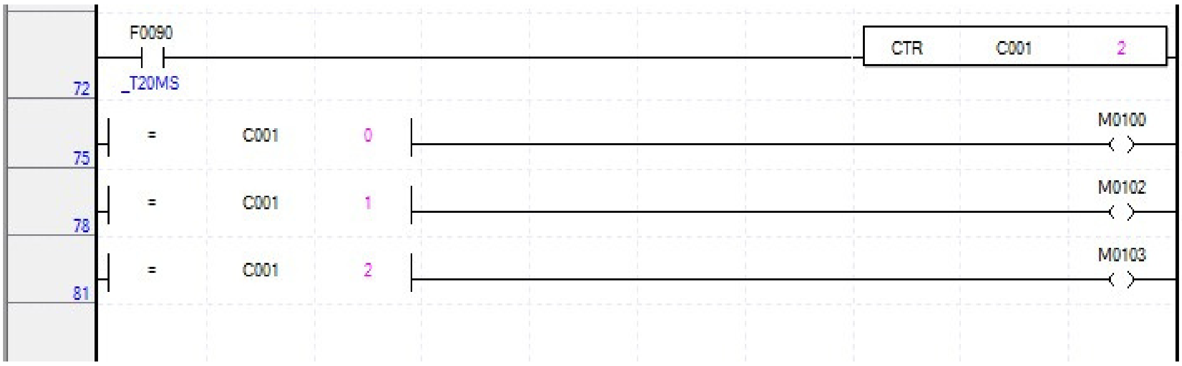
8- Double click on P2P Block:



Index	Ch.	Driver Setting	P2P function	Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number
0	2	Modbus RTU client	READ	M0100	Continuous	WORD	1	2	☑	1

9 - In index 0, configure:

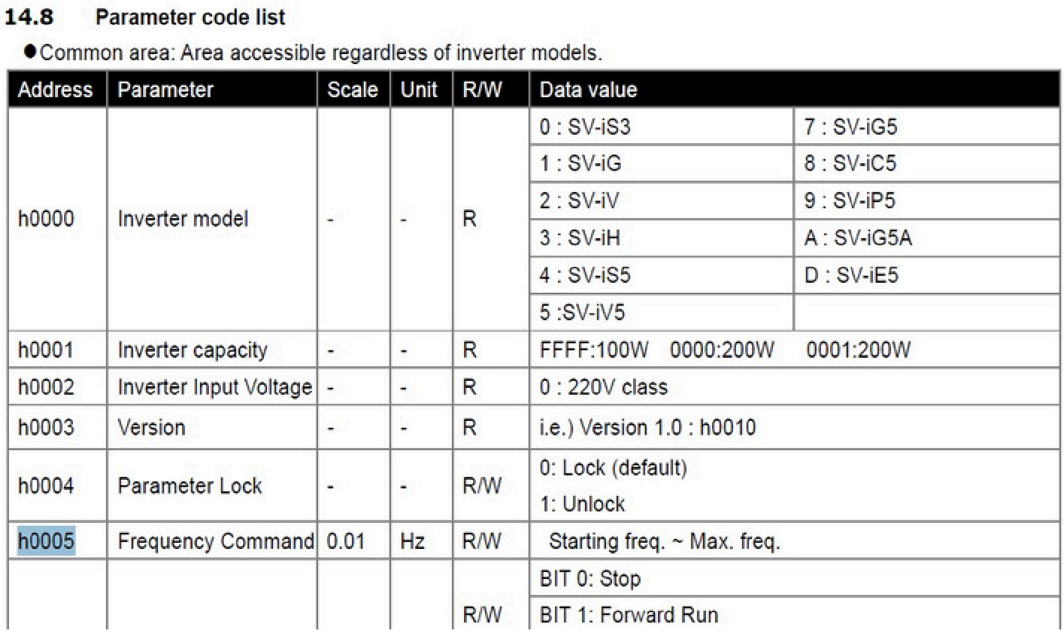
- Ch.: 2
- P2P Function: Choose, Read (Ler) / Write (Escrever)
- Conditional flag: M100 (for example) – In this field, you define which memory must be active to start communication. Example of programming to automatically activate this memory. The memories M100, M102, M103 correspond to the “Conditional Flag” of each line in the P2P Block.



- **Command Type: Continuous;**
- **Data Type: Word;**
- **Data Size: 2 – Number of words in sequence;**
- **Destination station number: 1 – This is the address configured on the Inverter. If you are working with more than one inverter, you can configure them in sequence: 1, 2, 3...**

10 – Click on Setting:

- **Read area: In this field, you will find the address that needs to be configured according to the specifications in the Inverter's manual:**

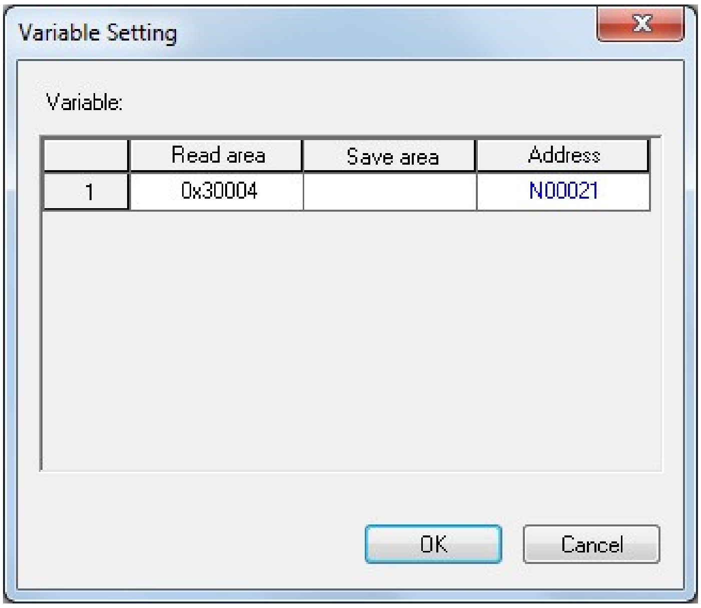


14.8 Parameter code list

●Common area: Area accessible regardless of inverter models.

Address	Parameter	Scale	Unit	R/W	Data value	
h0000	Inverter model	-	-	R	0 : SV-iS3	7 : SV-iG5
					1 : SV-iG	8 : SV-iC5
					2 : SV-iV	9 : SV-iP5
					3 : SV-iH	A : SV-iG5A
					4 : SV-iS5	D : SV-iE5
					5 :SV-iV5	
h0001	Inverter capacity	-	-	R	FFFF:100W 0000:200W 0001:200W	
h0002	Inverter Input Voltage	-	-	R	0 : 220V class	
h0003	Version	-	-	R	i.e.) Version 1.0 : h0010	
h0004	Parameter Lock	-	-	R/W	0: Lock (default) 1: Unlock	
h0005	Frequency Command	0.01	Hz	R/W	Starting freq. ~ Max. freq.	
				R/W	BIT 0: Stop	
					BIT 1: Forward Run	

Example: Depending on the PLC model, to configure the Frequency Command parameter, you need to configure the Read area as h0005 minus 1: 0x30004, as shown in the figure below:

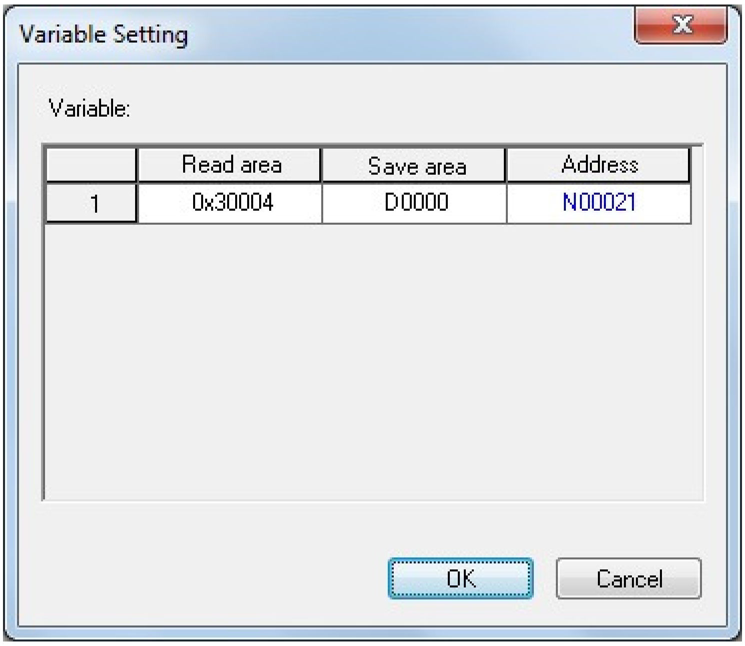


Variable Setting

Variable:

	Read area	Save area	Address
1	0x30004		N00021

Save area: In this field, you must choose which memory the read data will be sent to. In this example, we will choose memory D0.



Variable Setting

Variable:

	Read area	Save area	Address
1	0x30004	D0000	N00021

Remember that earlier we selected Data size as 2, which means that 2 words will be reserved in sequence. Since we chose D0000 to save the data from h0005, memory D0001 is also reserved, which will store the data from h0006. If the data size had been configured to 3, we could read the parameters h0005, h0006, h0007, and automatically save them in D0, D1, D2, and so on.

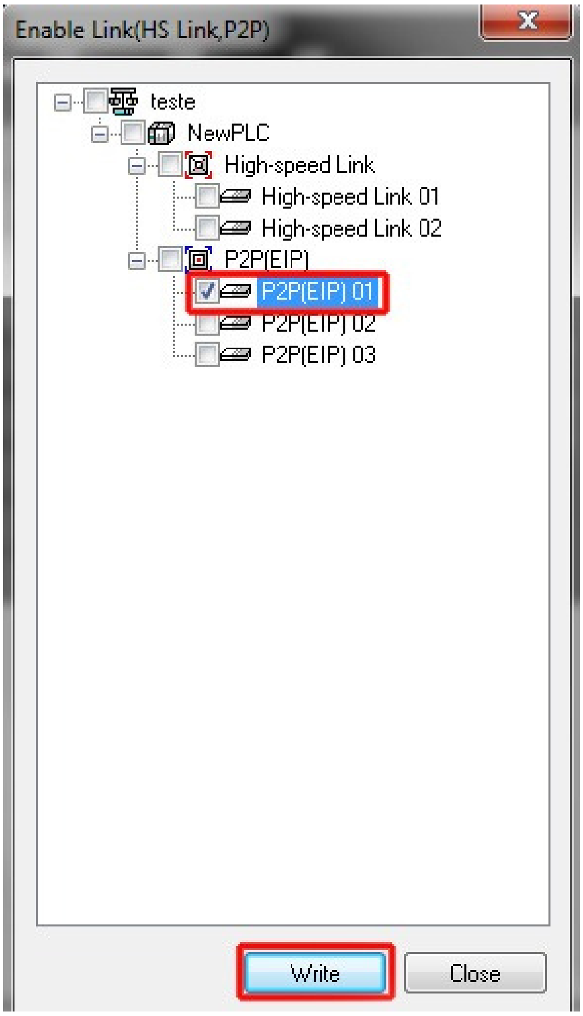
Now, you need to click, still in XG-PD, on:

On line > Connect;

On line > Write Parameter;

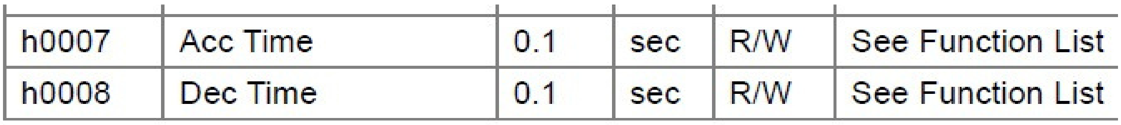
On line > Enable Link (HS Link, P2P);

Check the option P2P(EIP)01 and then click on Write:



Writing parameters to the Frequency Inverter:

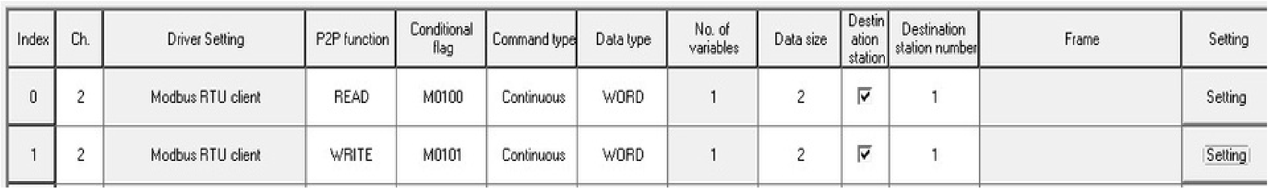
- **Example to change the acceleration and deceleration edge parameters of the inverter via the HMI.**



h0007	Acc Time	0.1	sec	R/W	See Function List
h0008	Dec Time	0.1	sec	R/W	See Function List

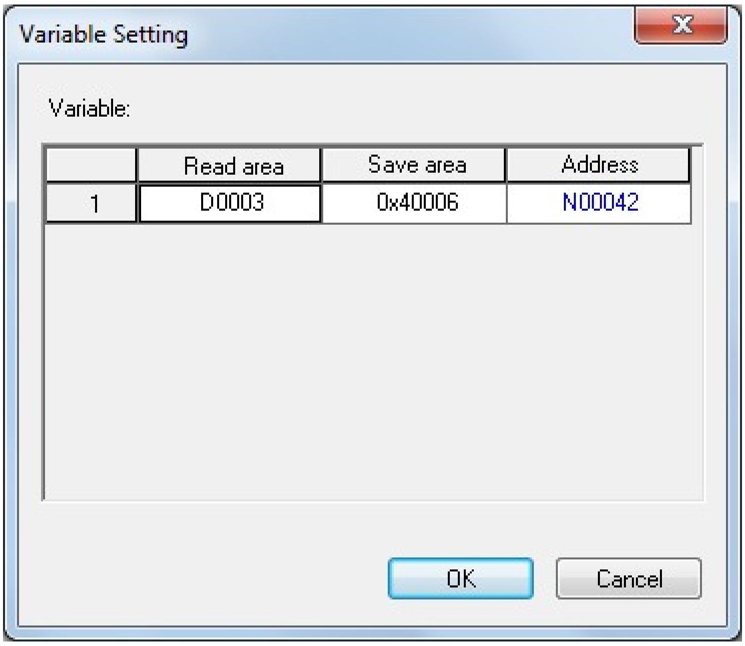
Figure: Inverter Manual.

Configure the P2P Block according to line 1 of the figure below:



Index	Ch.	Driver Setting	P2P function	Conditional flag	Command type	Data type	No. of variables	Data size	Destination station	Destination station number	Frame	Setting
0	2	Modbus RTU client	READ	M0100	Continuous	WORD	1	2	☑	1		Setting
1	2	Modbus RTU client	WRITE	M0101	Continuous	WORD	1	2	☑	1		Setting

O setting deve ser configurado da seguinte maneira: h0007 – 1: 0x40006



Variable:

	Read area	Save area	Address
1	D0003	0x40006	N00042

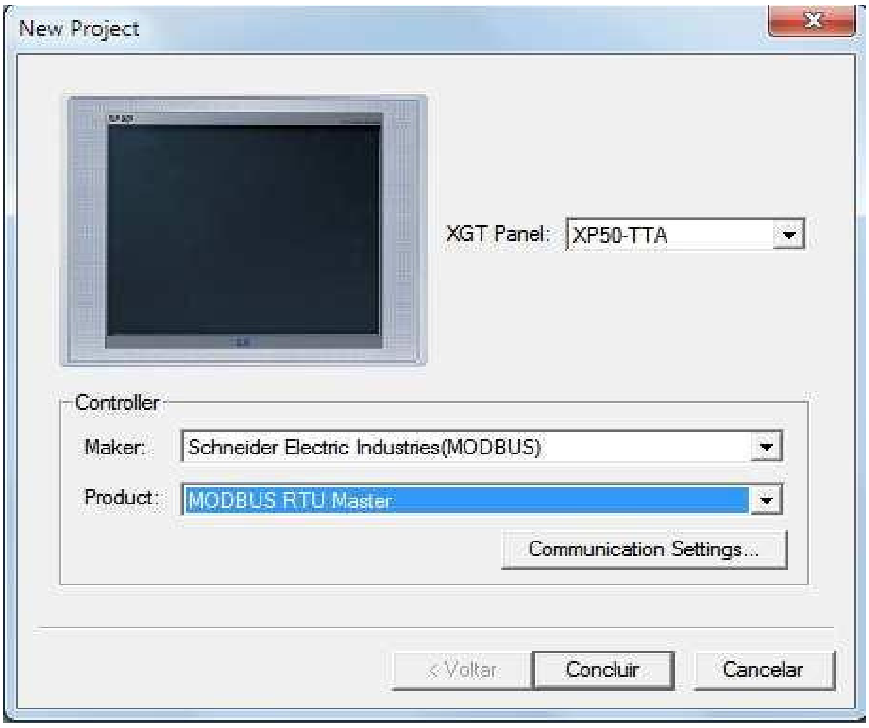
In this way, the data from h0007, which corresponds to acceleration, will need to be written to memory D3, and since the Data size was configured as 2, the data from h0008, which corresponds to deceleration, should be written to memory D4. In the HMI programming, you should create two Numeric Input fields and name them D3 and D4, respectively. When entering data into D3 or D4, it will automatically be written to the inverter.

And.

□WRITING VALUES INTO THE FREQUENCY INVERTER:

1- Open XP-Builder and configure the HMI slave device as shown in the screen below:

Modbus - RTU Communication: HMI with Frequency Inverter

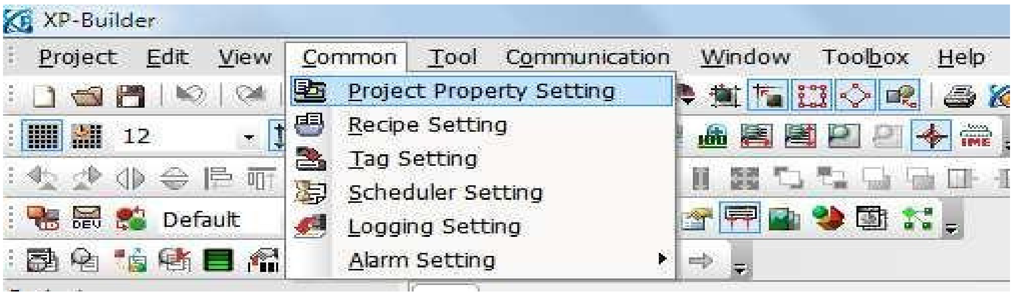


XGT Panel: XP50-TTA

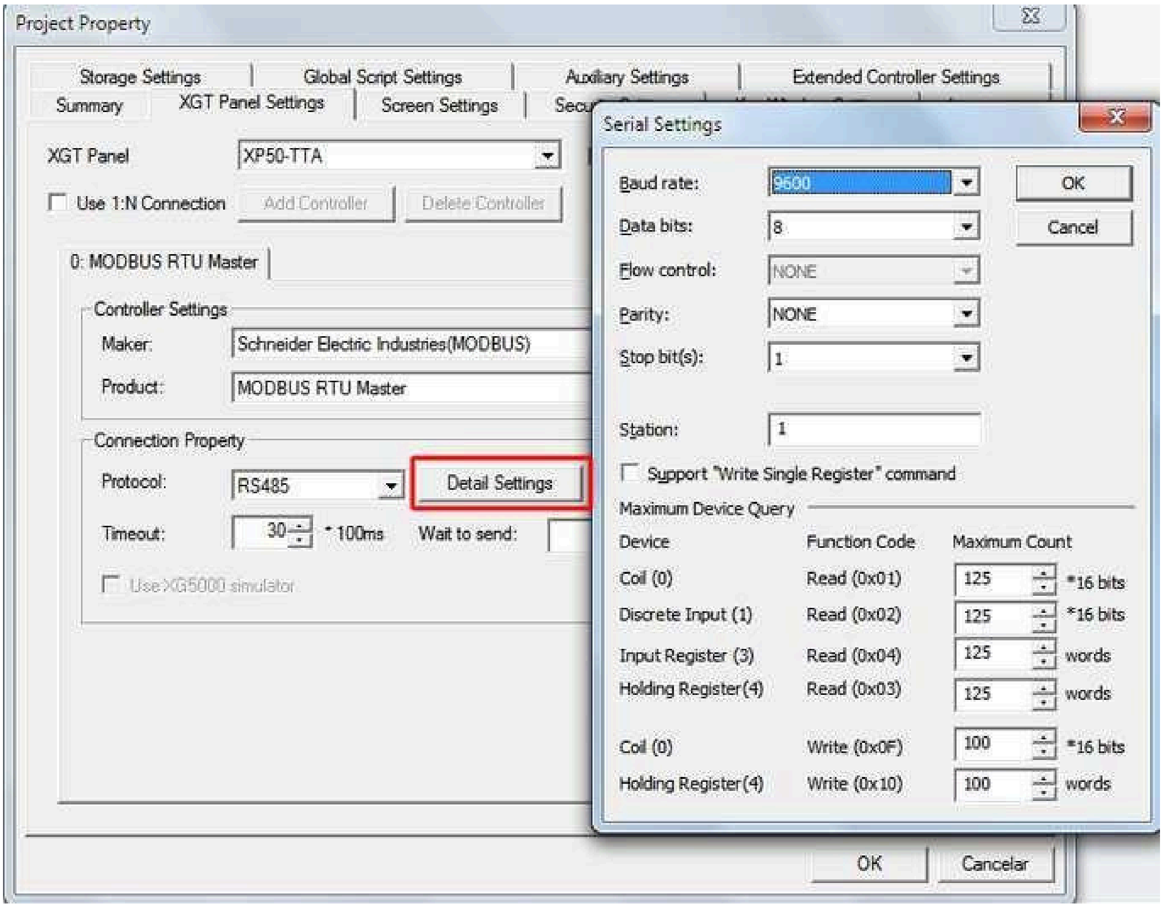
Maker: Schneider Electric Industries(MODBUS)

Product: MODBUS RTU Master

2- Click on Common > Project Property Settings



3- In the “XGT Panel Settings” tab we need to configure the communication medium (RS485) and the communication speed of the HMI with the Inverter, which in this case will be configured for both devices at 9600 bps:



XGT Panel: XP50-TTA

☐ Use 1:N Connection

0: MODBUS RTU Master

Controller Settings
Maker: Schneider Electric Industries(MODBUS)
Product: MODBUS RTU Master

Connection Property
Protocol: RS485
Timeout: 30 *100ms
Wait to send:

☐ Use XG5000 simulator

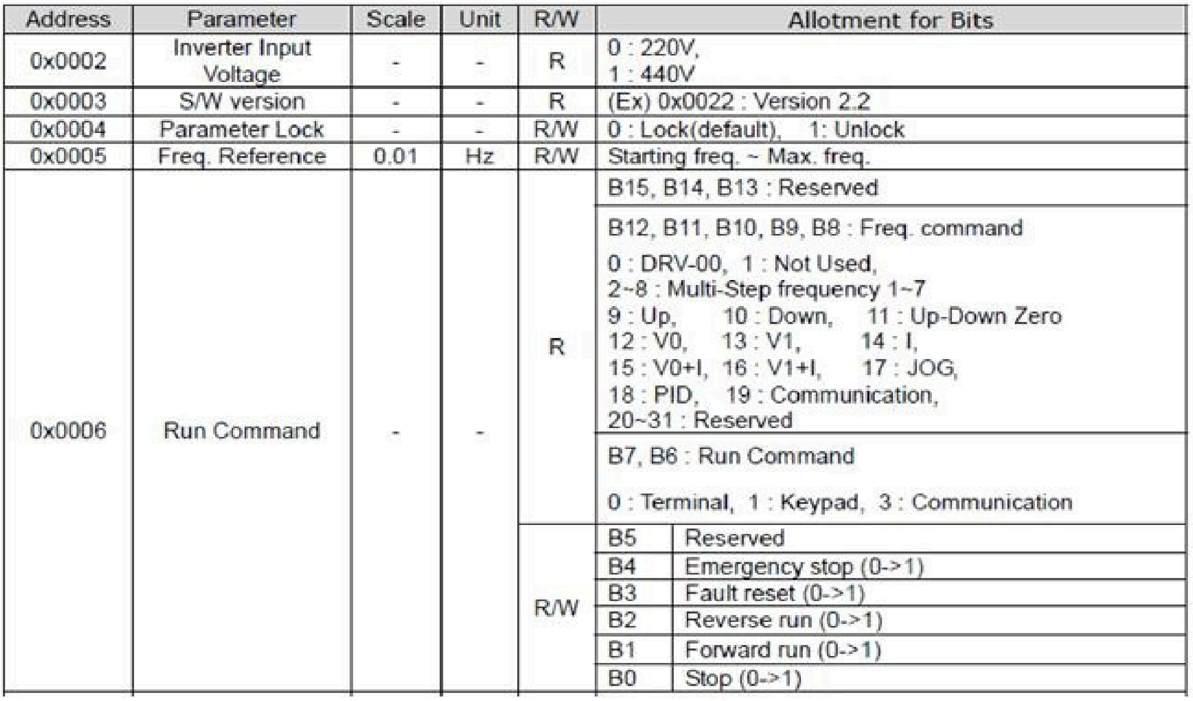
Serial Settings
Baud rate: 9600
Data bits: 8
Flow control: NONE
Parity: NONE
Stop bit(s): 1
Station: 1

☐ Support "Write Single Register" command

Maximum Device Query

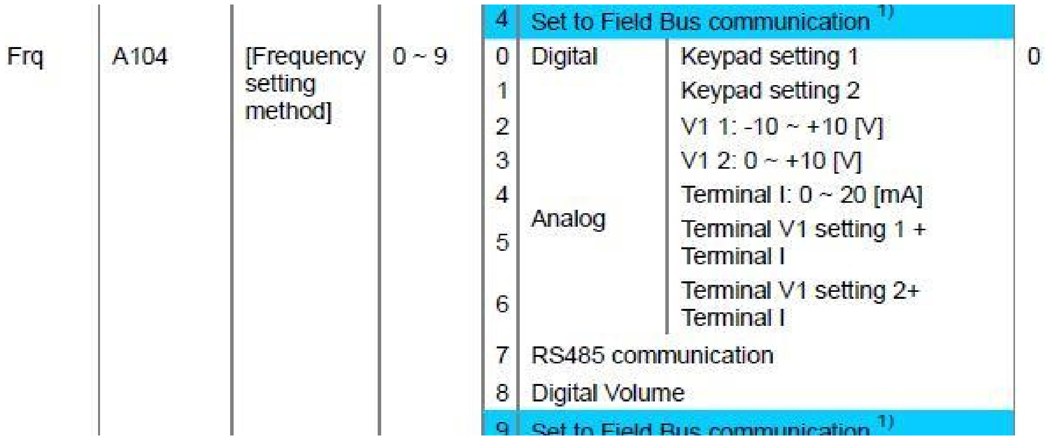
Device	Function Code	Maximum Count	
Coil (0)	Read (0x01)	125	*16 bits
Discrete Input (1)	Read (0x02)	125	*16 bits
Input Register (3)	Read (0x04)	125	words
Holding Register(4)	Read (0x03)	125	words
Coil (0)	Write (0x0F)	100	*16 bits
Holding Register(4)	Write (0x10)	100	words

4- After configuring the communication method between the HMI and the Inverter, we need to know which parameters we will write and read in the Frequency Inverter. In this case, we are using an LS inverter model IG5A, so below we will show a table that is contained in the inverter manual with the addresses of its parameters.



Address	Parameter	Scale	Unit	R/W	Allotment for Bits	
0x0002	Inverter Input Voltage	-	-	R	0 : 220V, 1 : 440V	
0x0003	S/W version	-	-	R	(Ex) 0x0022 : Version 2.2	
0x0004	Parameter Lock	-	-	R/W	0 : Lock(default), 1: Unlock	
0x0005	Freq. Reference	0.01	Hz	R/W	Starting freq. ~ Max. freq.	
0x0006	Run Command	-	-	R	B15, B14, B13 : Reserved	
					B12, B11, B10, B9, B8 : Freq. command 0 : DRV-00, 1 : Not Used, 2~8 : Multi-Step frequency 1~7 9 : Up, 10 : Down, 11 : Up-Down Zero 12 : V0, 13 : V1, 14 : I, 15 : V0+I, 16 : V1+I, 17 : JOG, 18 : PID, 19 : Communication, 20~31 : Reserved	
					B7, B6 : Run Command 0 : Terminal, 1 : Keypad, 3 : Communication	
				R/W	B5	Reserved
					B4	Emergency stop (0->1)
					B3	Fault reset (0->1)
					B2	Reverse run (0->1)
					B1	Forward run (0->1)
					B0	Stop (0->1)

5- In this example, we will first write a value in the Inverter Frequency parameter. To do this, we need to configure the frequency change method directly in the inverter. In the LS – IG5A Inverter, we have the following options:

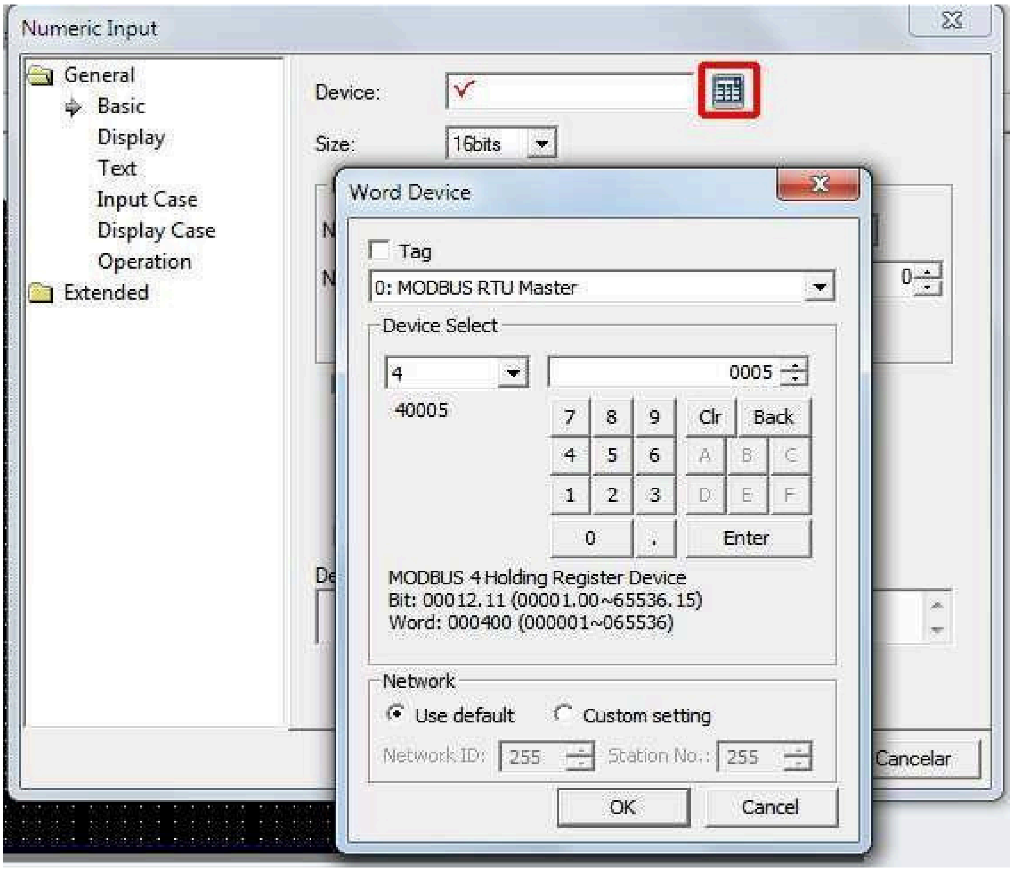


				4	Set to Field Bus communication 1)		
Frq	A104	[Frequency setting method]	0 ~ 9	0	Digital	Keypad setting 1	0
				1		Keypad setting 2	
				2	Analog	V1 1: -10 ~ +10 [V]	
				3		V1 2: 0 ~ +10 [V]	
				4		Terminal I: 0 ~ 20 [mA]	
				5		Terminal V1 setting 1 + Terminal I	
				6		Terminal V1 setting 2+ Terminal I	
				7	RS485 communication		
				8	Digital Volume		
				9	Set to Field Bus communication 1)		

In the LS inverter, select option number "7" within the "Frq" parameter.

Now that the inverter is configured to receive the frequency data from the HMI, we will create a Numeric Input on the HMI that will send the value we enter to the memory responsible for the frequency in the inverter.

6- In XP_Builder add a “Numeric Input” and configure it as shown in the screen below:



Device:

Size: 16bits

□ Tag

0: MODBUS RTU Master

Device Select

4 0005

40005

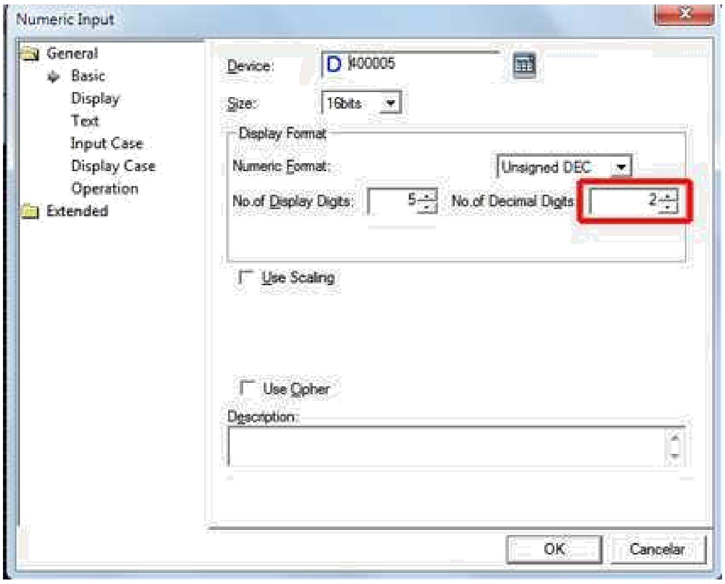
MODBUS 4 Holding Register Device
Bit: 00012.11 (00001.00~65536.15)
Word: 000400 (000001~065536)

Network

● Use default ○ Custom setting

Network ID: 255 Station No.: 255

Note that in the table presented above, the inverter memory that controls the frequency parameter is 0x0005. For this reason, we must enter the number “4” in the first field, which means “Write”, and in the second field the memory designated for Writing 0005. Click OK.



Device: D 400005

Size: 16bits

Display Format

Numeric Format: Unsigned DEC

No.of Display Digits: 5

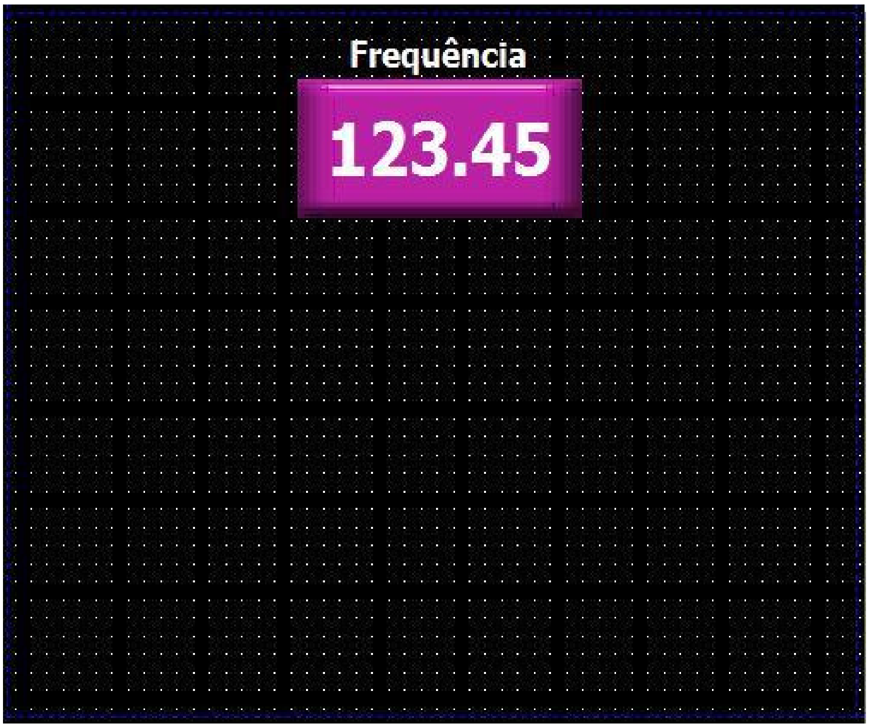
No.of Decimal Digits: 2

☐ Use Scaling

☐ Use Cipher

Description:

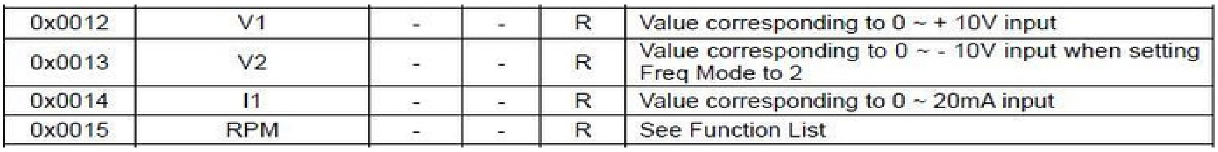
Place 2 decimal places after the decimal point for the value to appear correctly in the inverter.



Download the program and test.

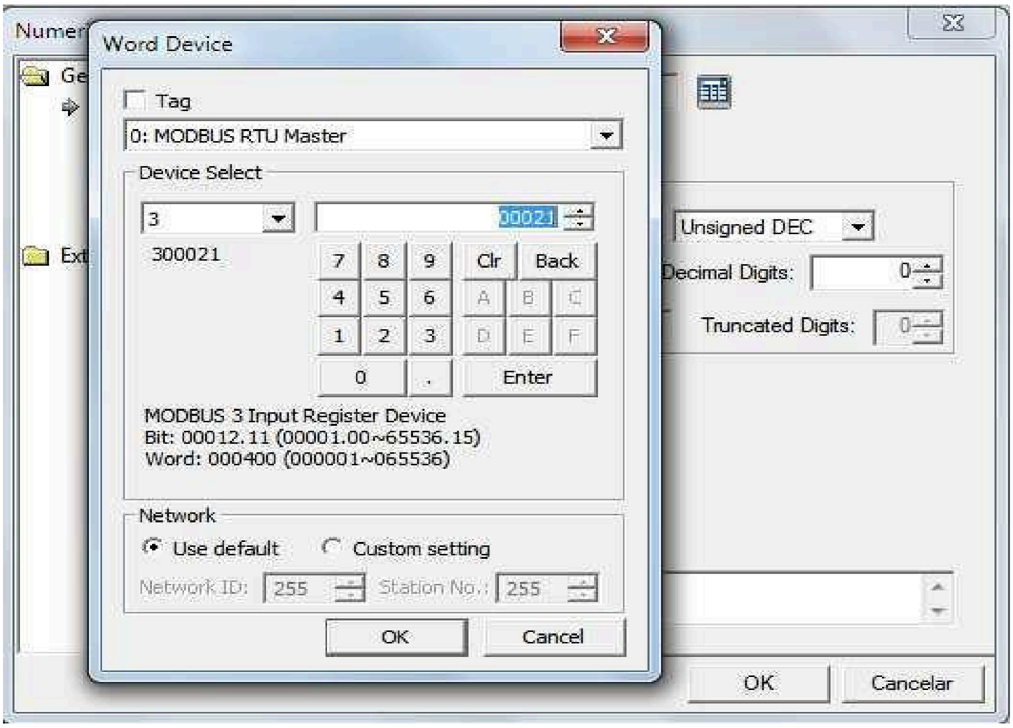
Reading Values from the Frequency Inverter:

7. Now, we will create a Numeric Display on the HMI to read a parameter from the frequency inverter. In this example, we will read the parameter that indicates the RPM value. According to the table in the inverter manual, the memory corresponding to the RPM parameter is 0x0015. This memory value is shown in hexadecimal in the manual, so we can convert it to decimal to input it into the HMI. Converting "15H" to decimal gives the value "21".



0x0012	V1	-	-	R	Value corresponding to 0 ~ + 10V input
0x0013	V2	-	-	R	Value corresponding to 0 ~ - 10V input when setting Freq Mode to 2
0x0014	I1	-	-	R	Value corresponding to 0 ~ 20mA input
0x0015	RPM	-	-	R	See Function List

8- In the first field, enter the number "3" for "Reading" and in the second field, memory 21 corresponding to the RPM parameter in decimal and click OK.



Word Device

□ Tag

0: MODBUS RTU Master

Device Select

3

00021

300021

MODBUS 3 Input Register Device
Bit: 00012.11 (00001.00~65536.15)
Word: 000400 (000001~065536)

Network

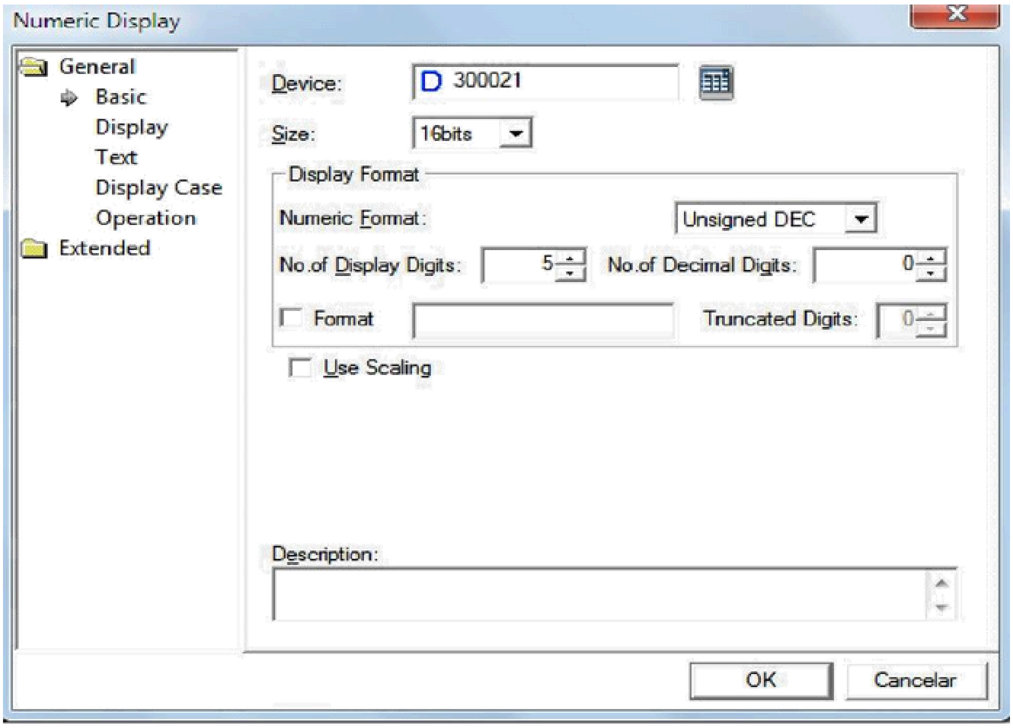
● Use default ○ Custom setting

Network ID: 255 Station No.: 255

Unsigned DEC

Decimal Digits: 0

Truncated Digits: 0



Numeric Display

Device: D 300021

Size: 16bits

Display Format

Numeric Format: Unsigned DEC

No.of Display Digits: 5

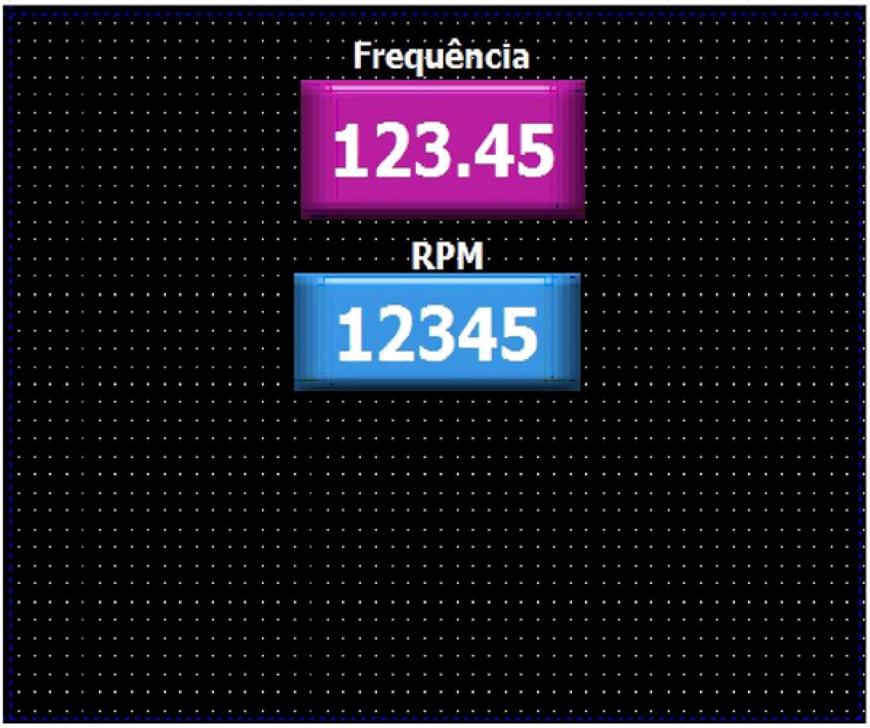
No.of Decimal Digits: 0

□ Format

Truncated Digits: 0

□ Use Scaling

Description:

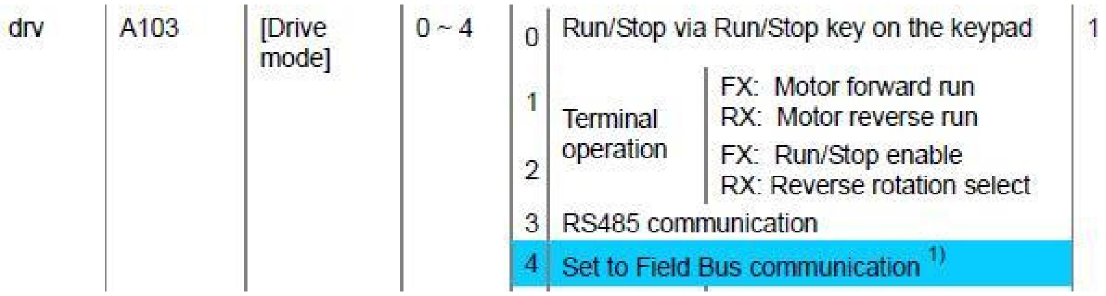


STARTING, REVERSING AND STOPPING THE MOTOR:

9- To start and stop the inverter we must configure the inverter starting mode. In the

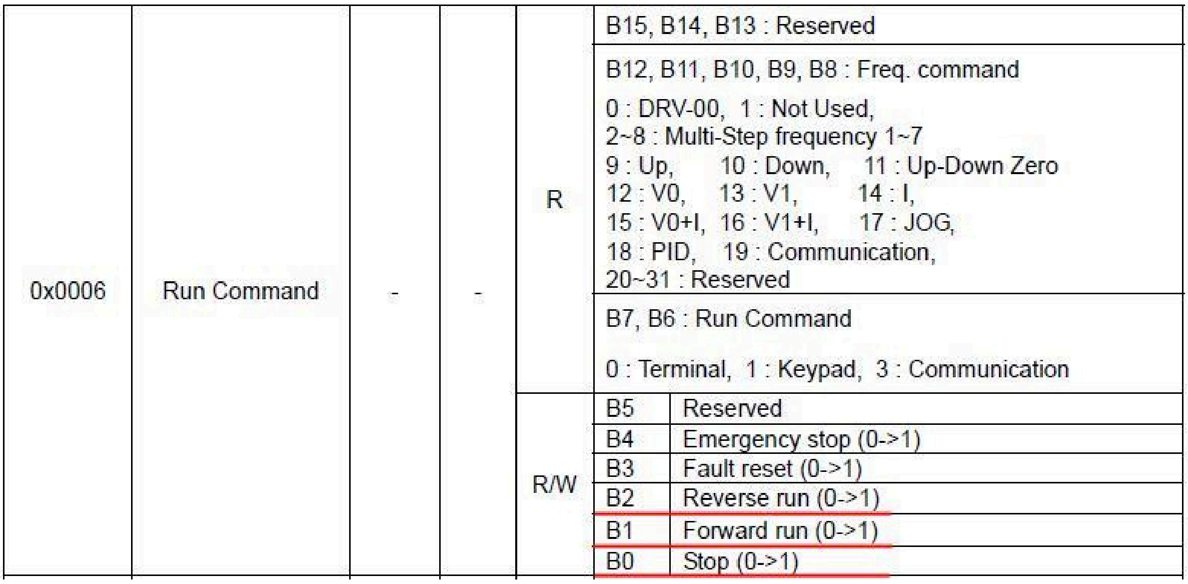
LS inverter the

DRV function must be changed to “3” according to the table below:



drv	A103	[Drive mode]	0 ~ 4	0	Run/Stop via Run/Stop key on the keypad		1
				1	Terminal operation	FX: Motor forward run RX: Motor reverse run	
				2		FX: Run/Stop enable RX: Reverse rotation select	
				3	RS485 communication		
				4	Set to Field Bus communication 1)		

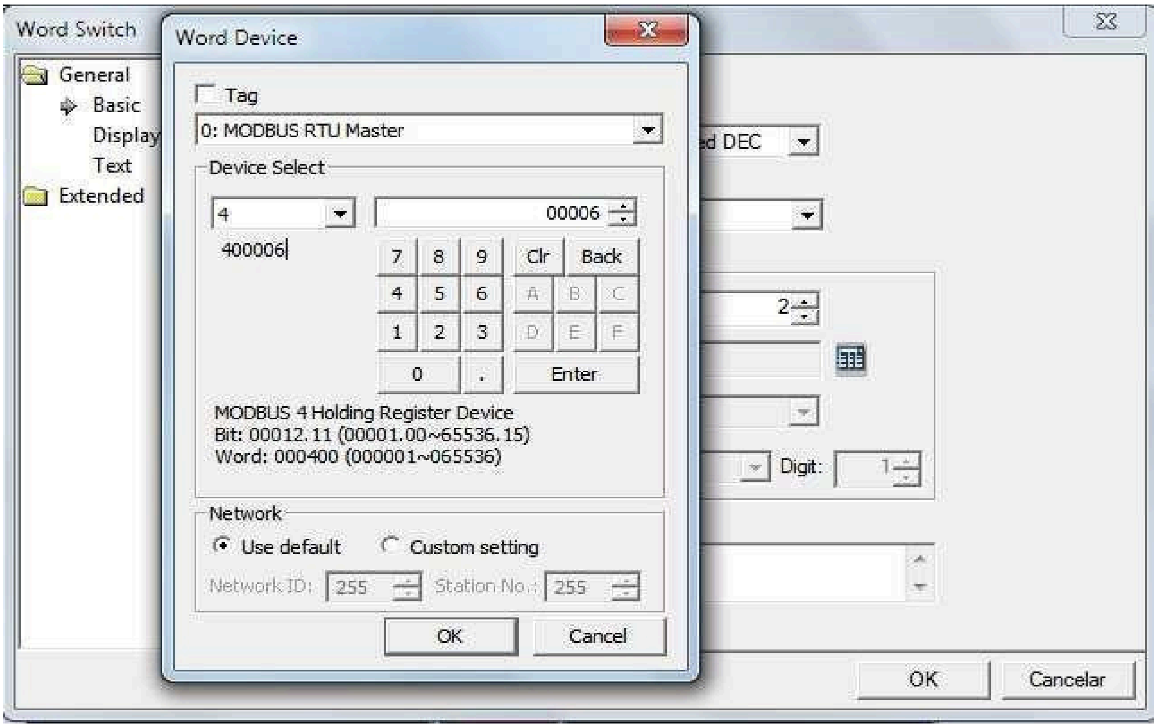
10- Now let's check the specific bits for the “Forward Run”, “Reverse Run” and “Stop” functions that are found in the same table seen previously:



0x0006	Run Command	-	-	R	B15, B14, B13 : Reserved	
					B12, B11, B10, B9, B8 : Freq. command 0 : DRV-00, 1 : Not Used, 2~8 : Multi-Step frequency 1~7 9 : Up, 10 : Down, 11 : Up-Down Zero 12 : V0, 13 : V1, 14 : I, 15 : V0+I, 16 : V1+I, 17 : JOG, 18 : PID, 19 : Communication, 20~31 : Reserved	
					B7, B6 : Run Command 0 : Terminal, 1 : Keypad, 3 : Communication	
				R/W	B5	Reserved
					B4	Emergency stop (0->1)
					B3	Fault reset (0->1)
					B2	Reverse run (0->1)
					B1	Forward run (0->1)
					B0	Stop (0->1)

We can see that bit 0 corresponds to Stop, bit 1 corresponds to Forward and bit 2 to Reverse.

11- In XP_Builder, create a "Word Switch" and configure the first field with the number "4" as the HMI will write this bit to the slave and in the second field, according to the table above, enter the Word "6".



Word Switch

Word Device

☐ Tag

0: MODBUS RTU Master

Device Select

4 | 00006

400006

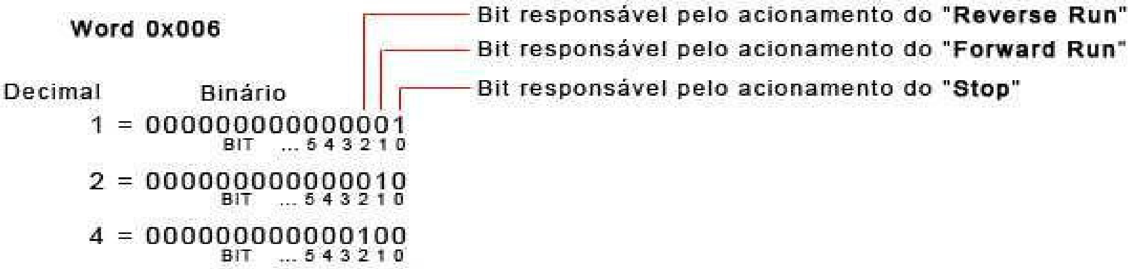
MODBUS 4 Holding Register Device
Bit: 00012.11 (00001.00~65536.15)
Word: 000400 (000001~065536)

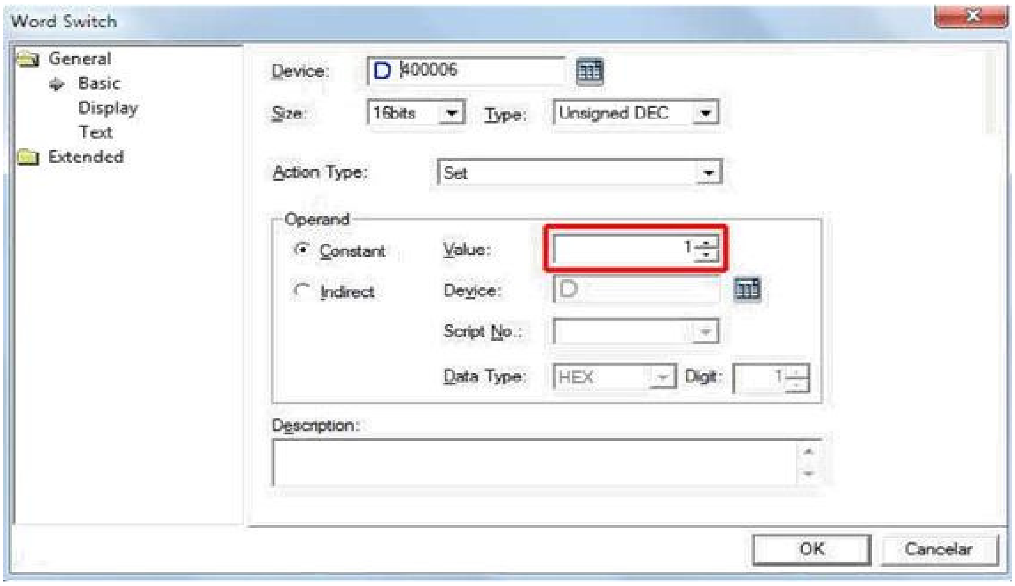
Network

● Use default ○ Custom setting

Network ID: 255 Station No.: 255

12- Now configure which bit of the Word you want to activate. First we want to activate the STOP bit which according to the table is bit 0. To activate bit 0 of a Word we need to write the number "1" in this Word. Example:





Word Switch

- General
  - Basic
  - Display
  - Text
- Extended

Device: D 400006

Size: 16bits　Type: Unsigned DEC

Action Type: Set

Operand

● Constant　Value: 1

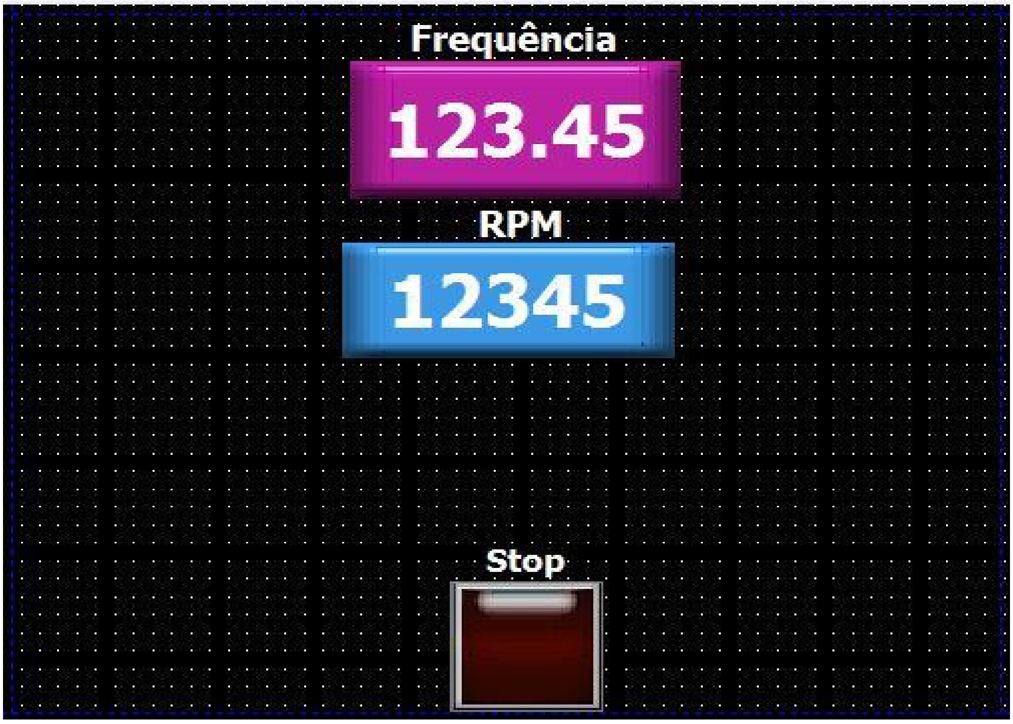
○ Indirect　Device: D

Script No.:

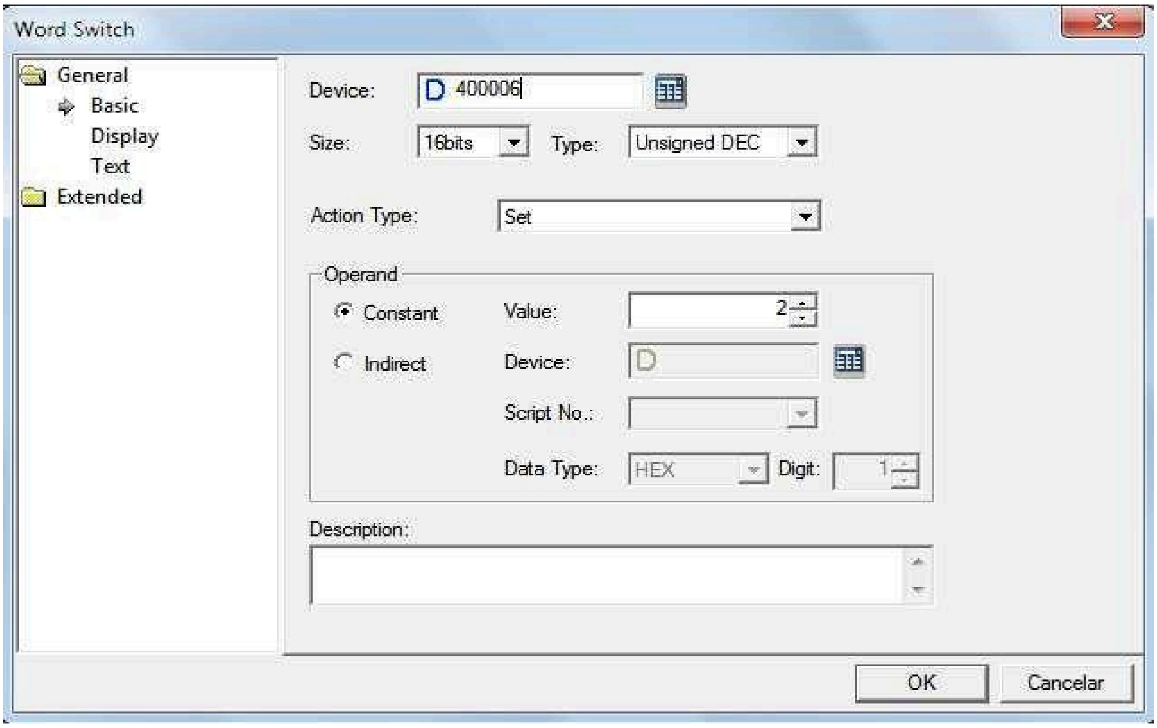
Data Type: HEX　Digit: 1

Description:

The figure above shows that when we press the Stop button, the number 1 will be sent to Word 0x006, activating bit 0 of this word.



13- To create the "Forward Run" button, click "Word Switch" again and configure the button as shown in the screen below:



Word Switch

General
Basic
Display
Text
Extended

Device: D 400006

Size: 16bits Type: Unsigned DEC

Action Type: Set

Operand

● Constant Value: 2

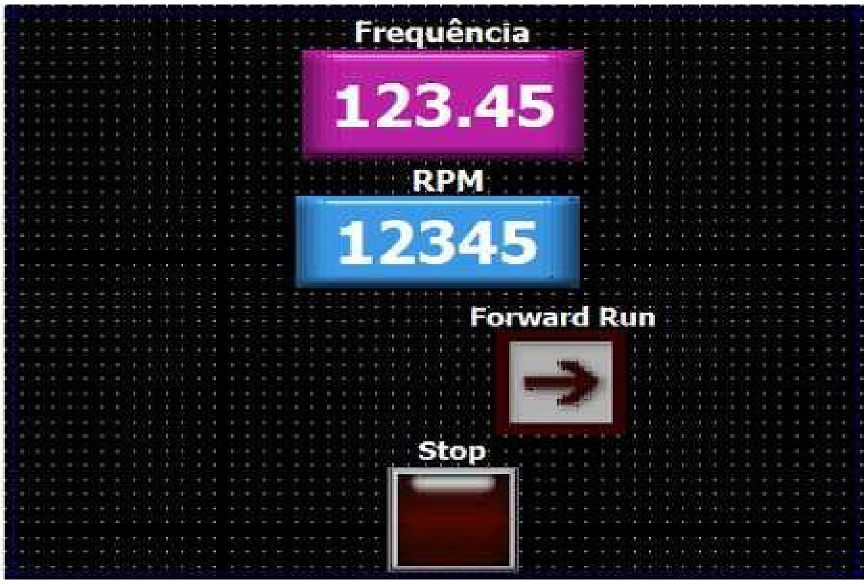
○ Indirect Device: D

Script No.:

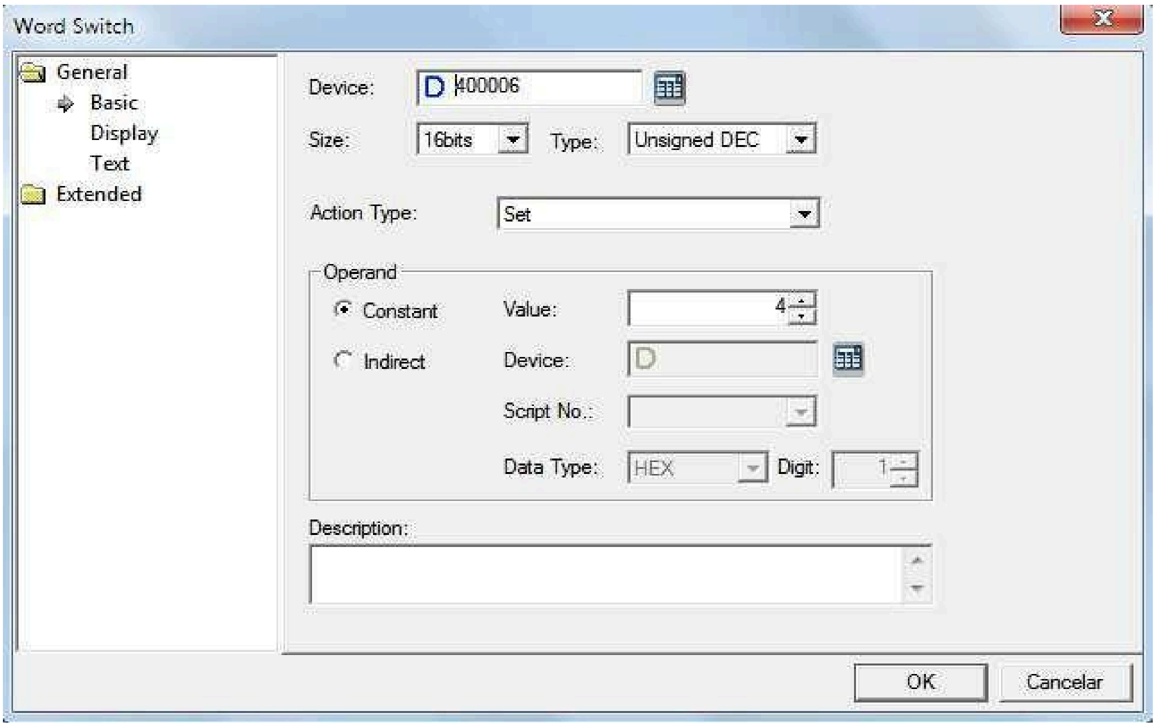
Data Type: HEX Digit: 1

Description:

The figure above shows that when we press the Forward Run button, the number 2 will be sent to Word 0x006, activating bit 1 of this word.



14- Finally, let's create the “Reverse Run” button. Click on “Word Switch” again and configure it as shown in the screen below:



Device: D 400006

Size: 16bits Type: Unsigned DEC

Action Type: Set

Operand

● Constant Value: 4

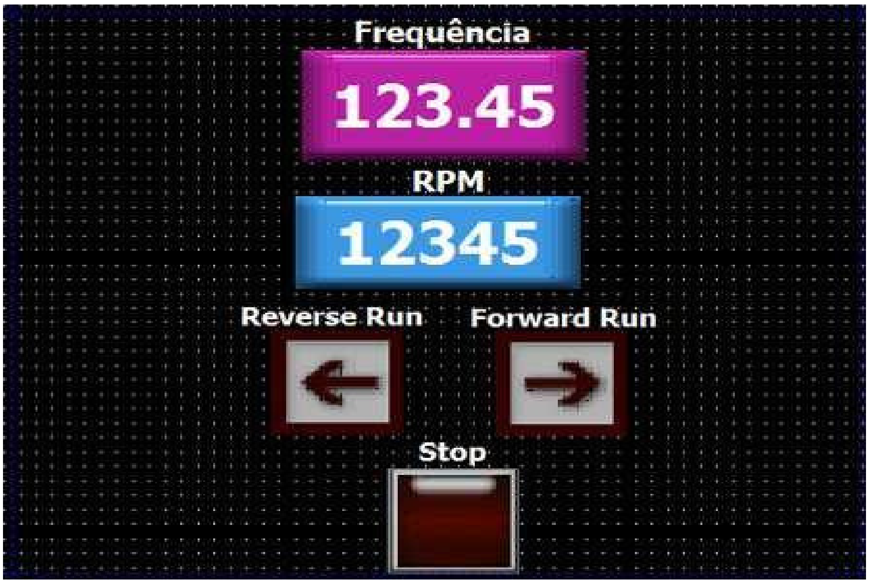
○ Indirect Device: D

Script No.:

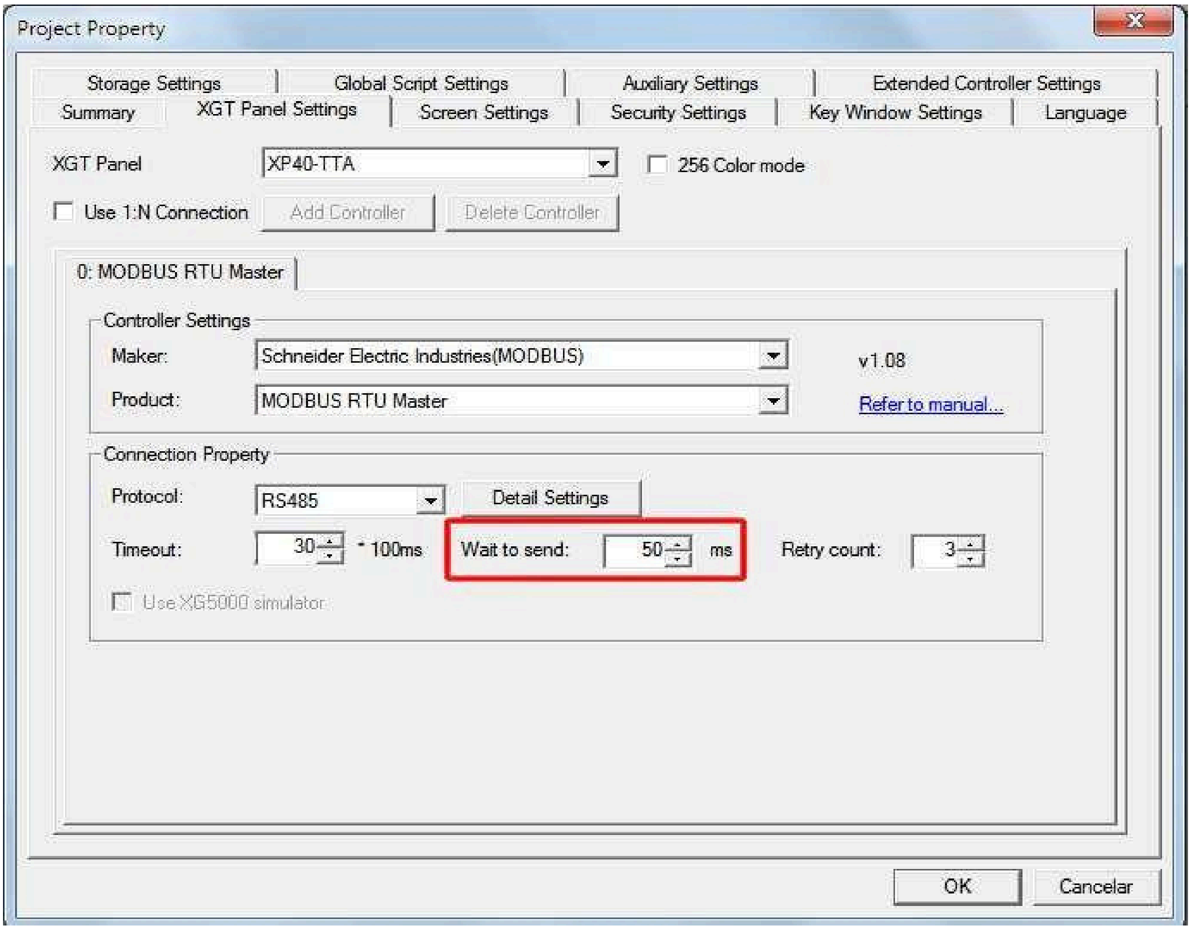
Data Type: HEX Digit: 1

Description:

The figure above shows that when we press the Reverse Run button, the number 4 will be sent to Word 0x006, activating bit 2 of this word.



If communication between the inverter and the HMI is slow, configure “Wait to send” as shown in the screen below:



XGT Panel: XP40-TTA

□ 256 Color mode

□ Use 1:N Connection

Maker: Schneider Electric Industries(MODBUS)

Product: MODBUS RTU Master

v1.08

Protocol: RS485

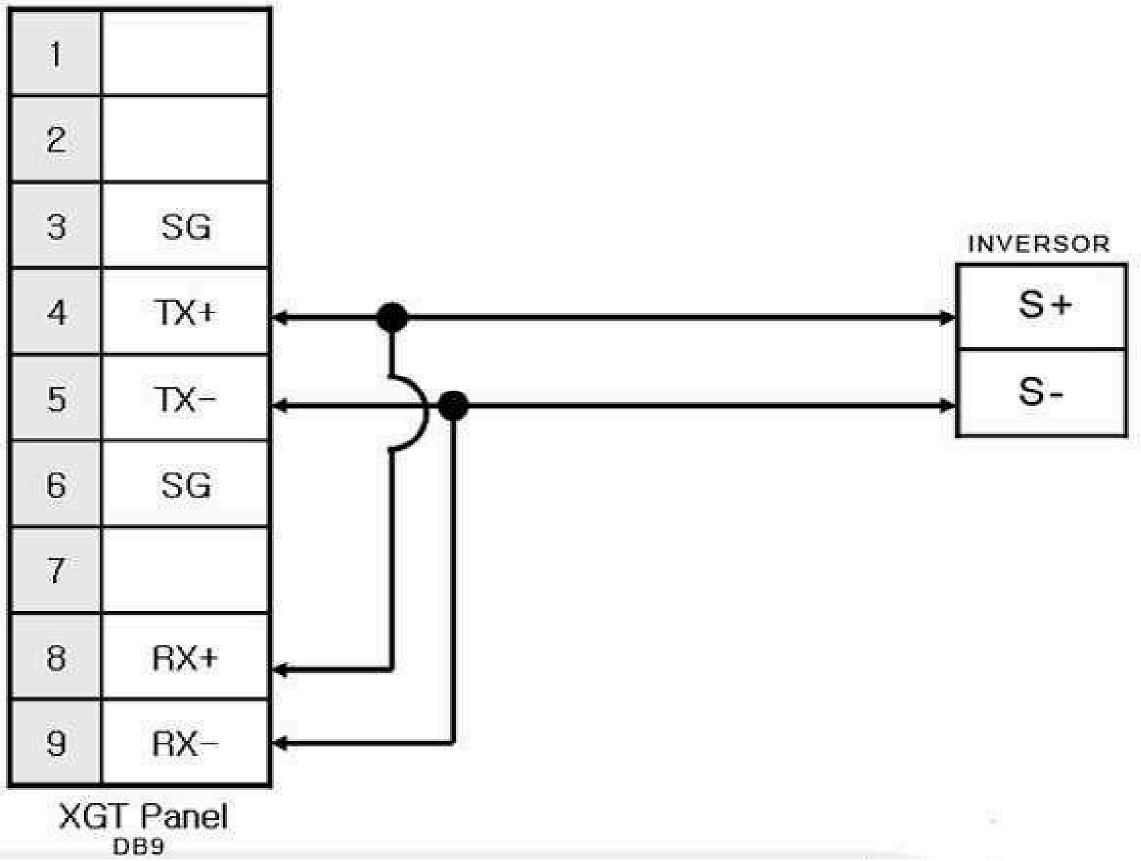
Timeout: 30 * 100ms

Wait to send: 50 ms

Retry count: 3

□ Use XG5000 simulator

RS485 communication cable connection diagram

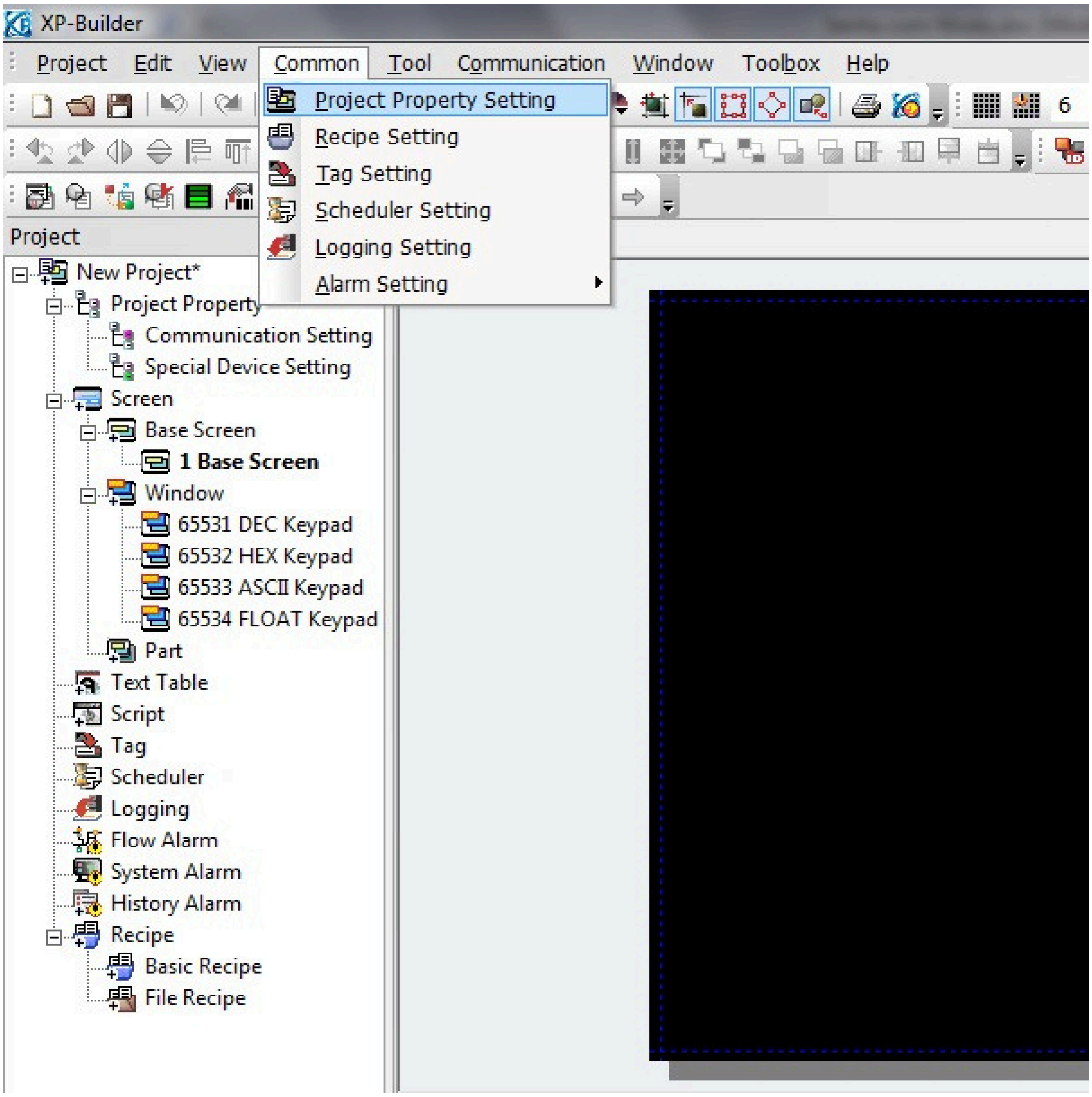


And.

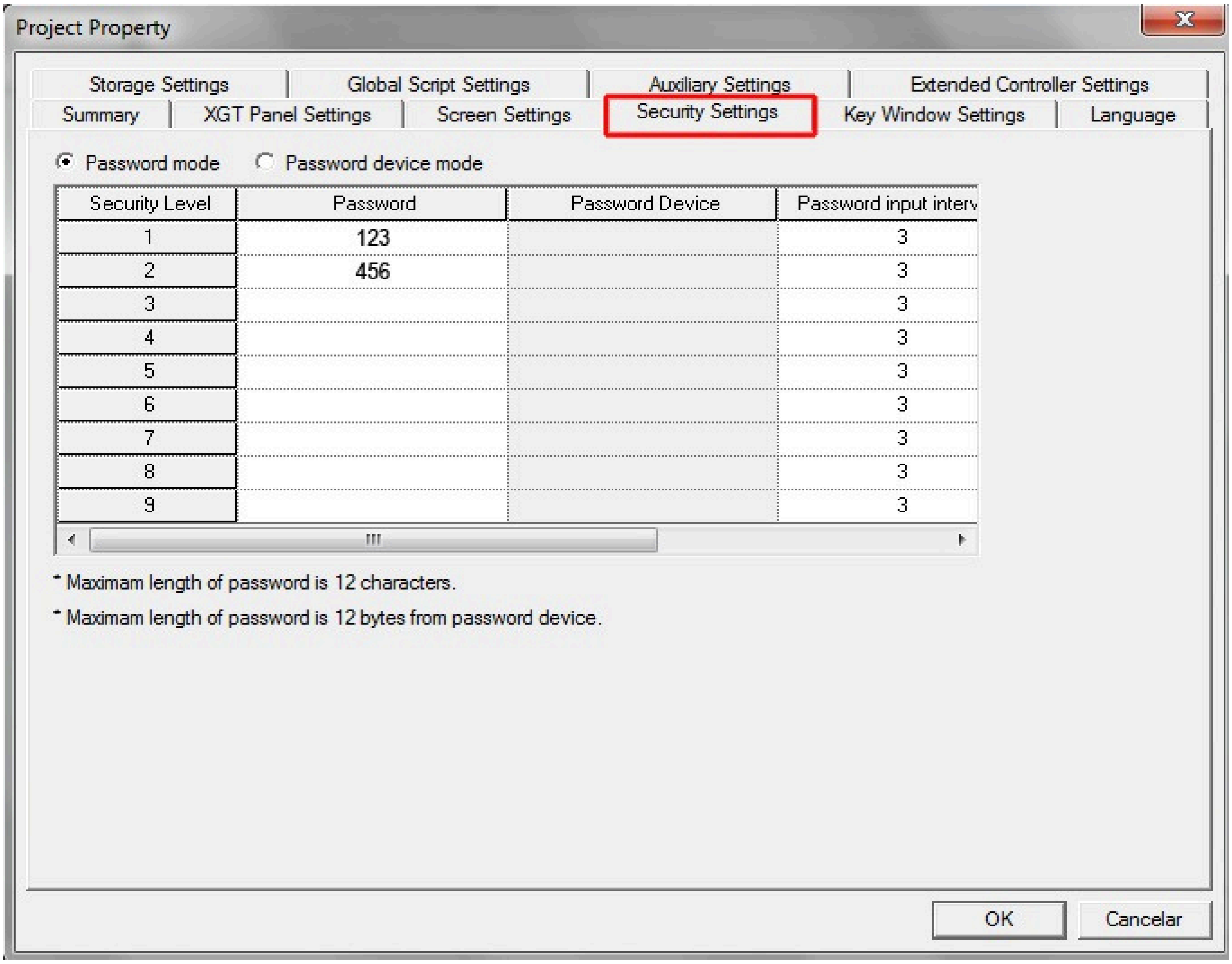
Password with levels in HMI

□

In XP-Builder, click on Common > Project Property Setting:



In the Security Settings tab, we can enter passwords according to the desired levels. In Password input interval, we can enter the time that the password will remain active for the user to make the necessary changes to the program. This means that if the user has entered the password correctly, during these 3 minutes the password does not need to be entered again.



Project Property

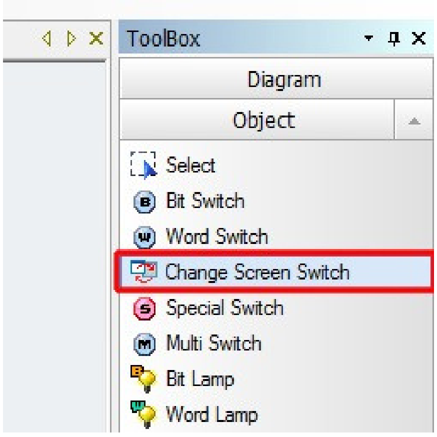
● Password mode ○ Password device mode

Security Level	Password	Password Device	Password input interv
1	123		3
2	456		3
3			3
4			3
5			3
6			3
7			3
8			3
9			3

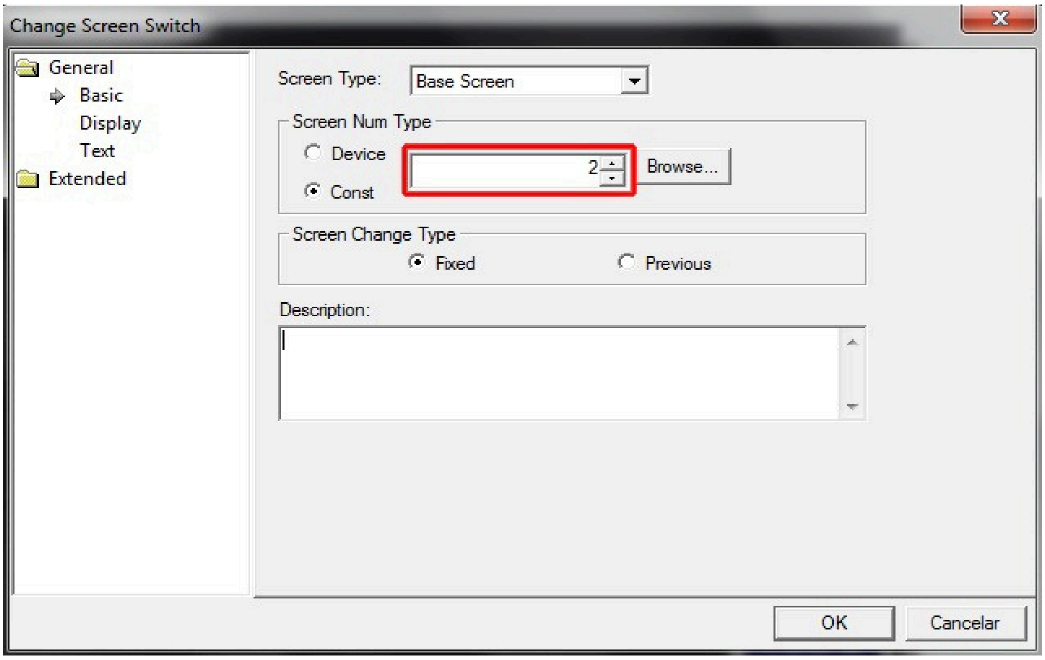
* Maximam length of password is 12 characters.
* Maximam length of password is 12 bytes from password device.

To test this function, we will create a screen change button, but now with the passwords already configured, the user will need to enter this password to move from one screen to another.

Click on Change Screen Switch and create a button on the screen:



Enter the screen number for which you want the change to occur.


Change Screen Switch

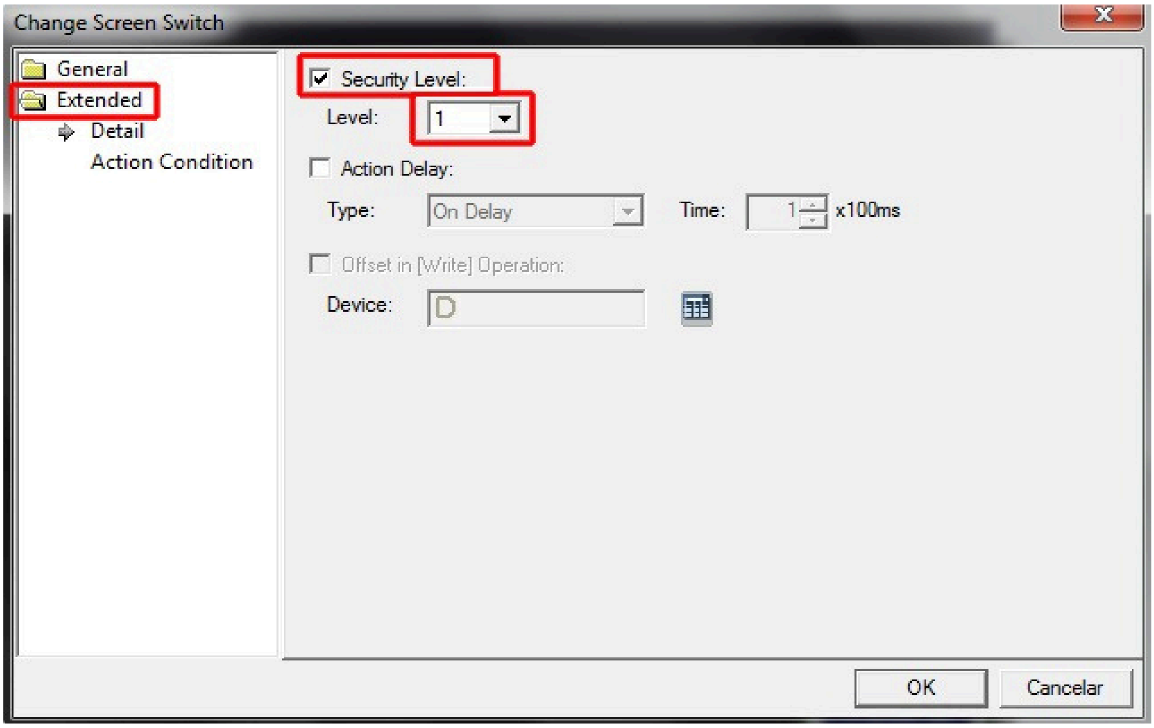
Screen Type: Base Screen

Screen Num Type: ○ Device ● Const 2

Screen Change Type: ● Fixed ○ Previous

Description:

Click on Extended and select the Security Level option and in Level define which Level will be used for screen 2 of the program. In this case, we will leave level 1, which has the numerical sequence 1, 2 and 3 as the password. It is in the Extended function that we find the option to enter a password for all commands.



Change Screen Switch

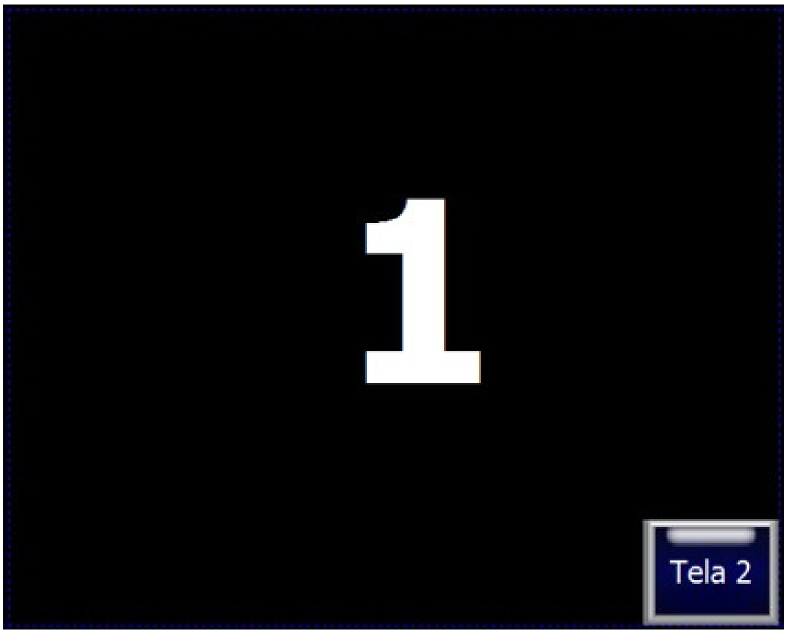
General
Extended
Detail
Action Condition

☑ Security Level:
Level: 1

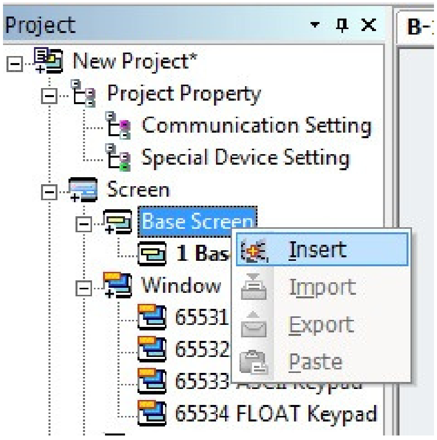
□ Action Delay:
Type: On Delay Time: 1 x100ms

□ Offset in [Write] Operation:
Device: D

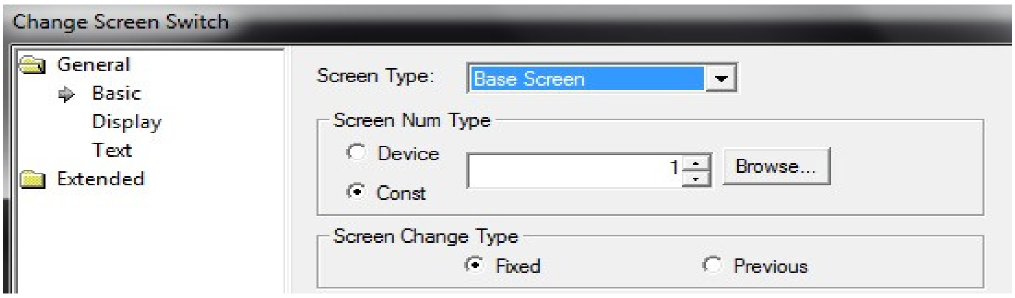
OK Cancelar



To create a new screen, right-click on Base Screen and then click on Insert:



Configure the screen change button in the same way, but now to switch to screen 1 and with password level 2:


Change Screen Switch

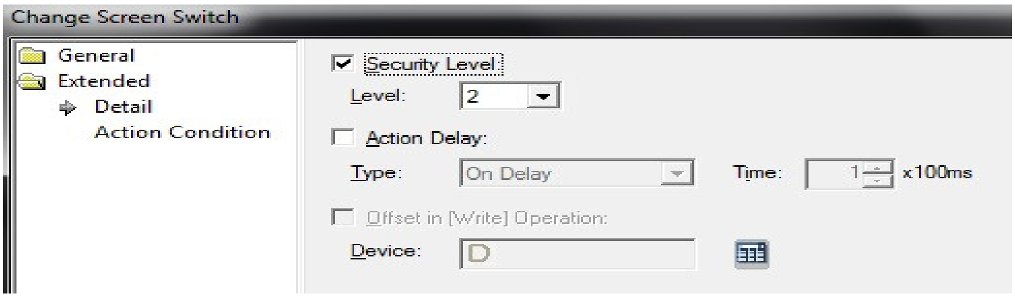
Screen Type: Base Screen

Screen Num Type

○ Device
● Const 1

Screen Change Type

● Fixed ○ Previous


Change Screen Switch

☑ Security Level

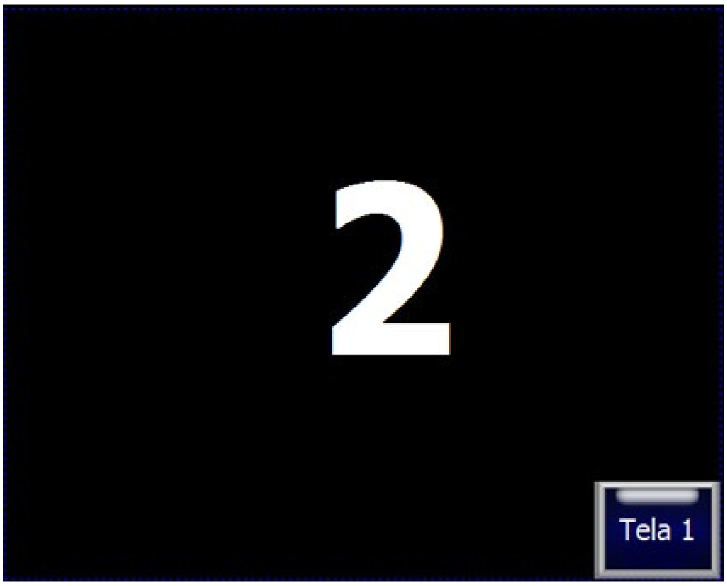
Level: 2

□ Action Delay:

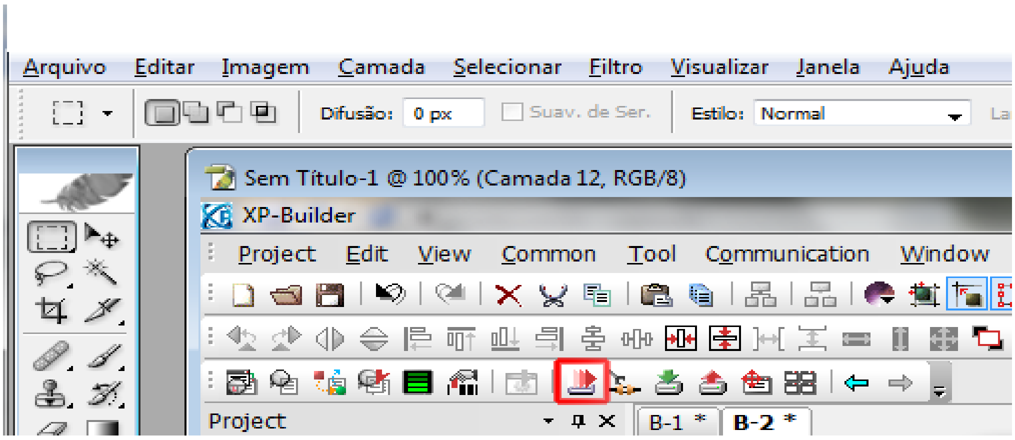
Type: On Delay Time: 1 x100ms

□ Offset in [Write] Operation:

Device: D



We can simulate the program to confirm its operation:


Arquivo Editar Imagem Camada Selecionar Filtro Visualizar Janela Ajuda
Difusão: 0 px
Suav. de Ser.
Estilo: Normal
Sem Título-1 @ 100% (Camada 12, RGB/8)
XP-Builder
Project Edit View Common Tool Communication Window
Project
B-1 *
B-2 *

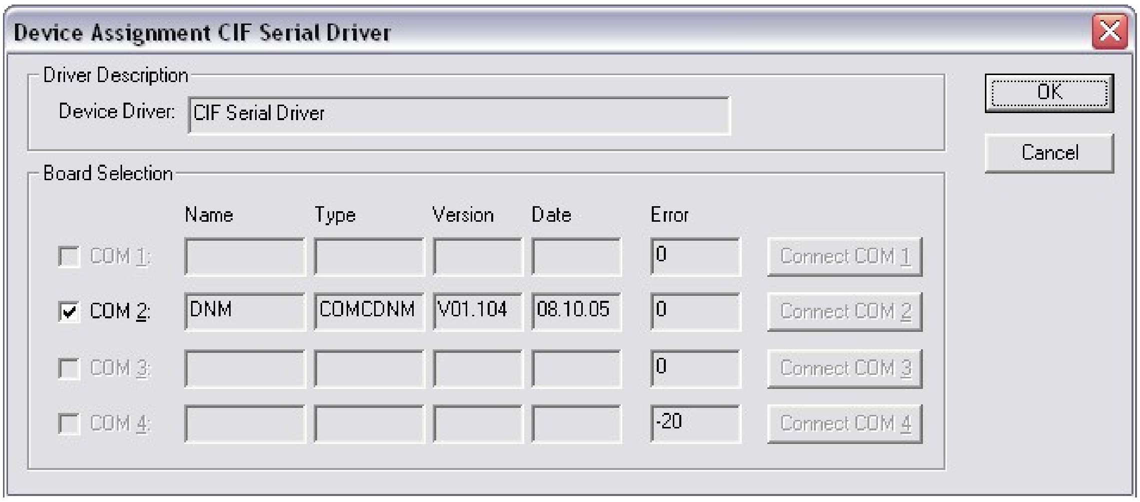
When we click to change to screen two, the program will ask for the level 1 password (123) and to return to screen 2, the program will ask for the level 2 password (345).

And.

DeviceNet Communication Tutorial

- Open SYCON > click on new;
- Select DeviceNet > ok;
- Click on Insert > Master;
- In Available masters select the device and click on Add > ok;
- Select the newly created master;
- Click on Setting > Master Setting > Setting > Check "Buffered, host controlled" > ok > ok;
- Click on Setting > Device Assingment > click Connect COM 1,2... > Check the desired COM
as shown in the screen below > ok:

DeviceNet Communication



Device Assignment CIF Serial Driver

Driver Description

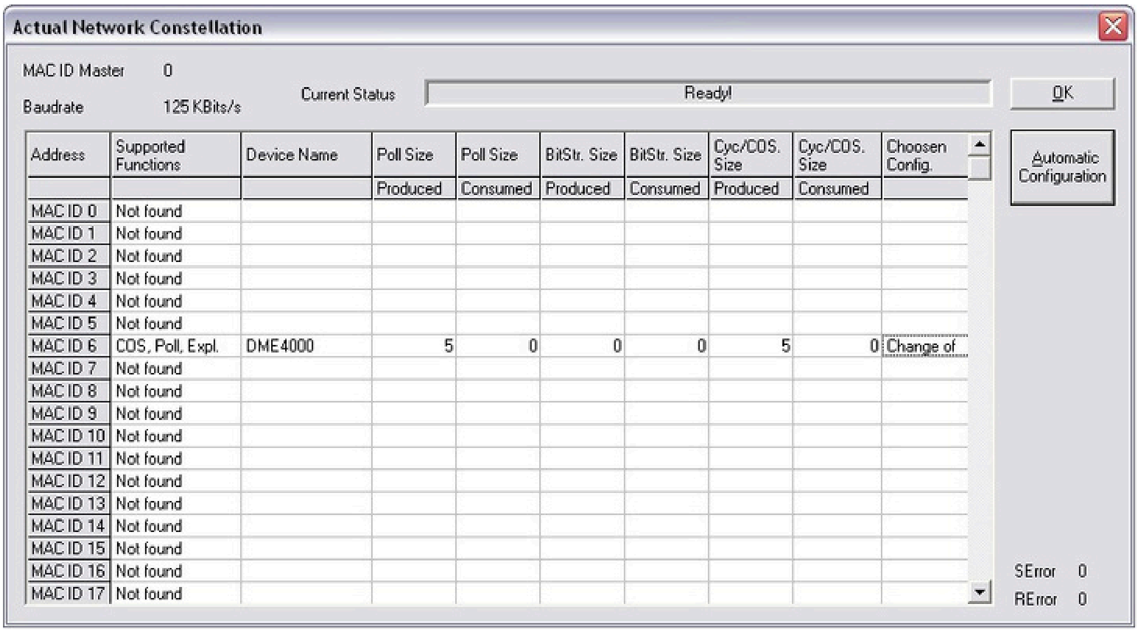
Device Driver: CIF Serial Driver

Board Selection

	Name	Type	Version	Date	Error	
□ COM 1:					0	Connect COM 1
☑ COM 2:	DNM	COMCDNM	V01.104	08.10.05	0	Connect COM 2
□ COM 3:					0	Connect COM 3
□ COM 4:					-20	Connect COM 4

- Click Settings > Bus Parameters > check Baudrate (125Kbits/s) > click OK;

- Click Online > Automatic Network Scan. Wait for the screen to load and it will look like this:



Actual Network Constellation

MAC ID Master 0

Baudrate 125 KBits/s

Current Status Ready!

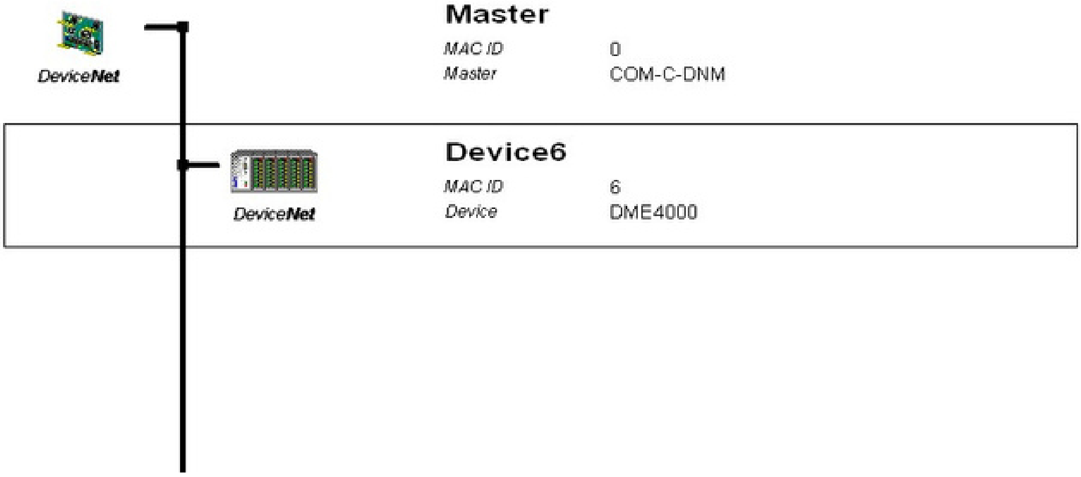
Address	Supported Functions	Device Name	Poll Size	Poll Size	BitStr. Size	BitStr. Size	Cyc/COS. Size	Cyc/COS. Size	Choosen Config.
			Produced	Consumed	Produced	Consumed	Produced	Consumed	
MAC ID 0	Not found								
MAC ID 1	Not found								
MAC ID 2	Not found								
MAC ID 3	Not found								
MAC ID 4	Not found								
MAC ID 5	Not found								
MAC ID 6	COS, Poll, Expl.	DME4000	5	0	0	0	5	0	Change of
MAC ID 7	Not found								
MAC ID 8	Not found								
MAC ID 9	Not found								
MAC ID 10	Not found								
MAC ID 11	Not found								
MAC ID 12	Not found								
MAC ID 13	Not found								
MAC ID 14	Not found								
MAC ID 15	Not found								
MAC ID 16	Not found								
MAC ID 17	Not found								

SError 0

RError 0

- On this same screen, click on Automatic Configuration > yes > ok;

- The screen will be as follows:



Master

MAC ID 0

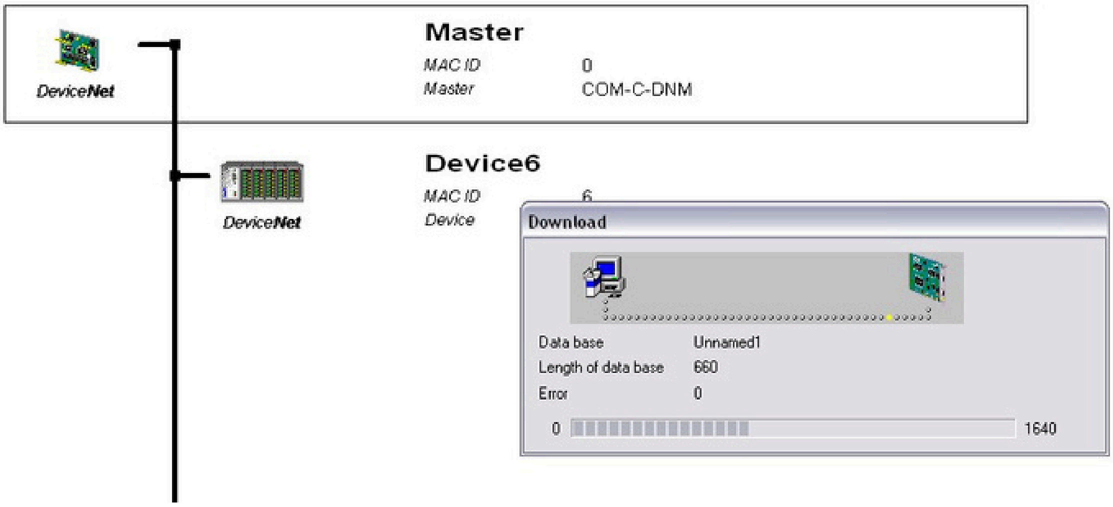
Master COM-C-DNM

Device6

MAC ID 6

Device DME4000

- Select the Master and click Online > Download > Yes;



MAC ID 0
Master COM-C-DNM

MAC ID 6
Device

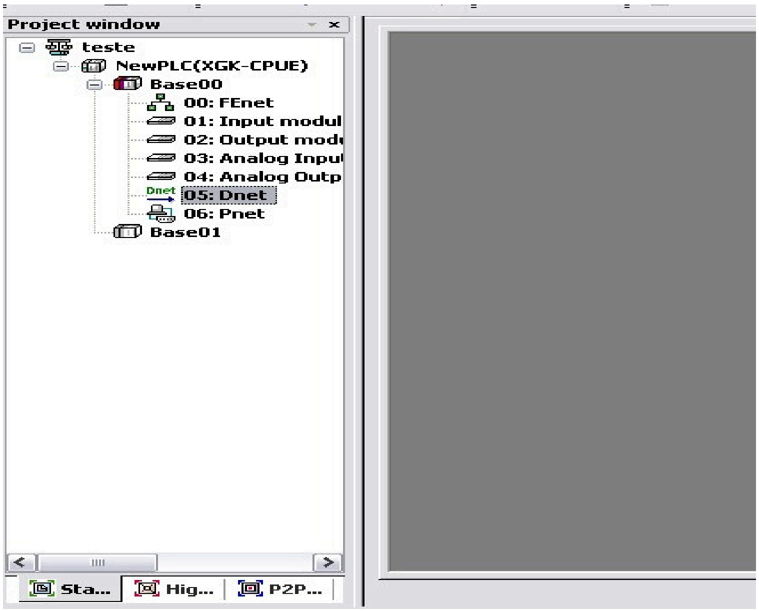
Data base Unnamed1
Length of data base 660
Error 0

- Now open XGPD (in XG5000 > Tools > Network Manager);

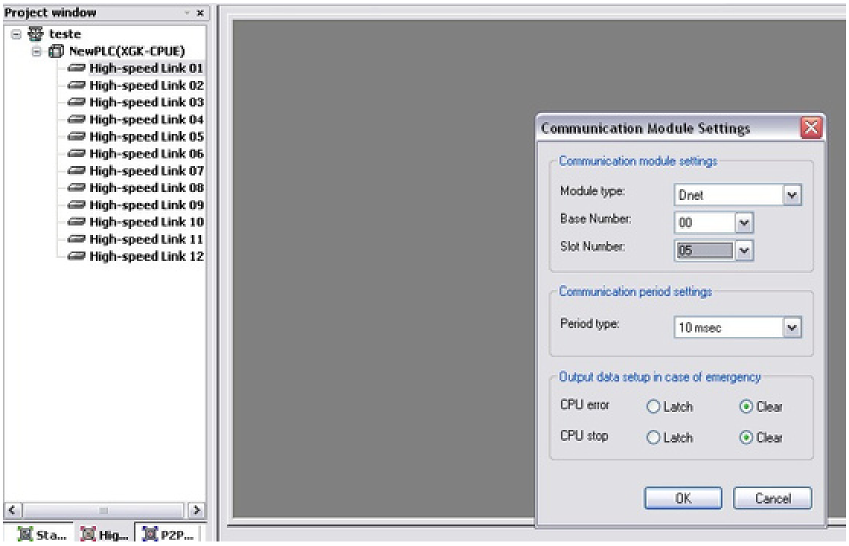
- New > File name > Select PLC > Select PLC Model > ok;

- Click on Online > Connect;

- Click on Online > Read IO Information:



- Click on the High-speed Link tab > Double-click High-speed Link 01 > In Module Type select Dnet > in Slot Number select the slot number where the Devinet module is located, as shown in the screen below and click ok:



Communication Module Settings

Communication module settings

Module type: Dnet

Base Number: 00

Slot Number: 05

Communication period settings

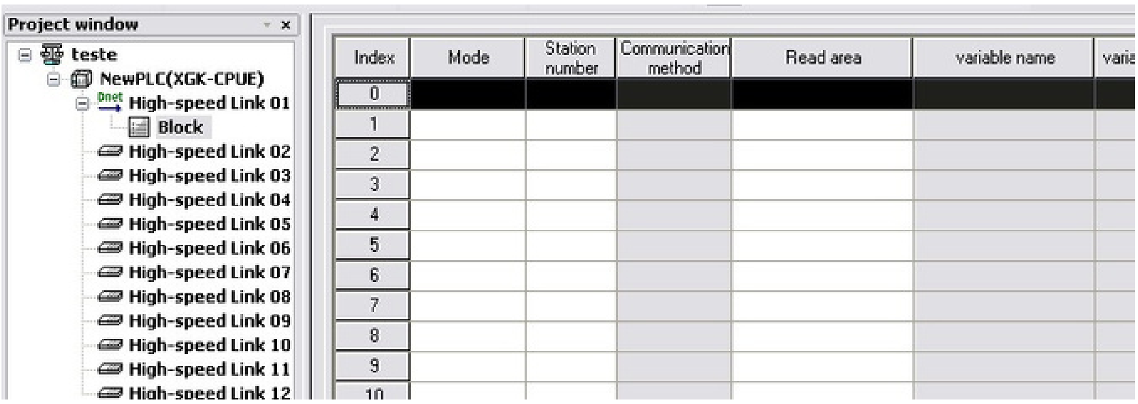
Period type: 10 msec

Output data setup in case of emergency

CPU error ○ Latch ● Clear

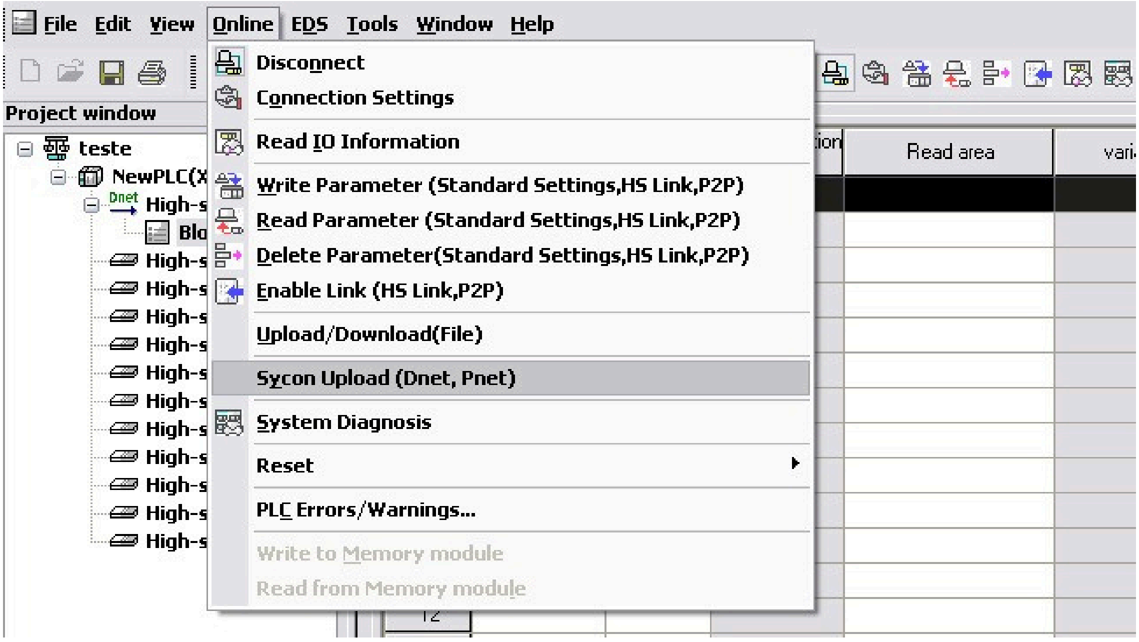
CPU stop ○ Latch ● Clear

- A table with several indexes will open, click once on Block and then click once on 0 in the first row as shown in the figure:

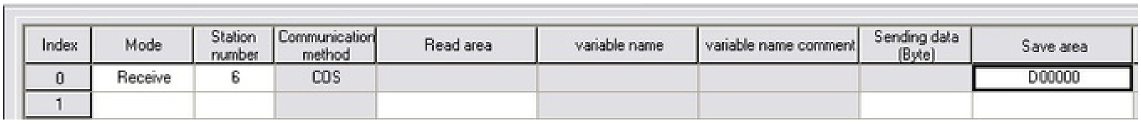


Index	Mode	Station number	Communication method	Read area	variable name	varia
0						
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

- Now click on Online > Sycon Upload:

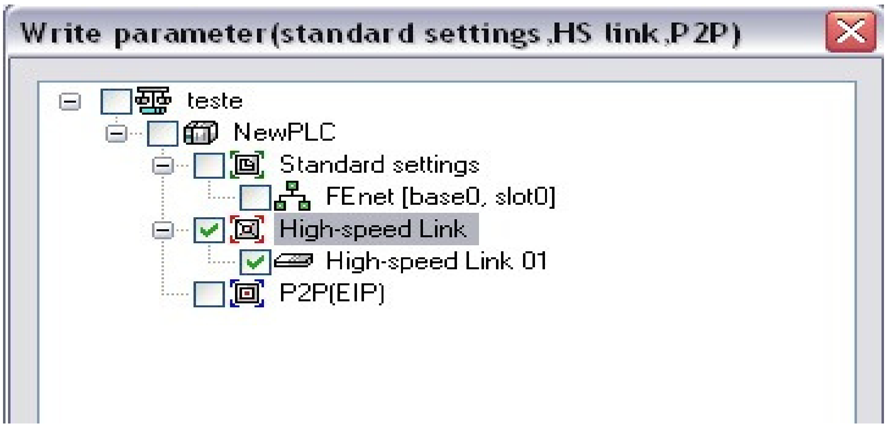

File Edit View Online EDS Tools Window Help
Project window
teste
NewPLC(X
High-s
Blo
Disconnect
Connection Settings
Read IO Information
Write Parameter (Standard Settings,HS Link,P2P)
Read Parameter (Standard Settings,HS Link,P2P)
Delete Parameter(Standard Settings,HS Link,P2P)
Enable Link (HS Link,P2P)
Upload/Download(File)
Sycon Upload (Dnet, Pnet)
System Diagnosis
Reset
PLC Errors/Warnings...
Write to Memory module
Read from Memory module
Read area

- The program will fill in index 0 automatically, you just need to fill in which memory will be used to receive the data, in this case we address memory D0, as can be seen in the screen below:



Index	Mode	Station number	Communication method	Read area	variable name	variable name comment	Sending data (Byte)	Save area
0	Receive	6	COS					D00000
1								

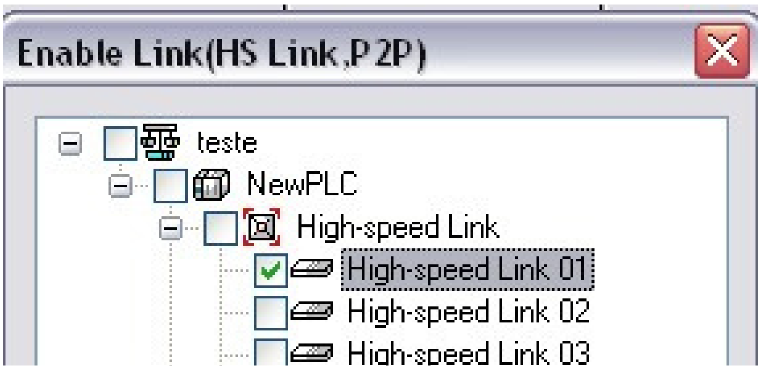
- Click on Online > Write Parameter and check the option as shown in the screen below and click ok:



Write parameter(standard settings,HS link,P2P)

- □ teste
  - □ NewPLC
    - □ Standard settings
      - □ FEnet [base0, slot0]
    - ☑ High-speed Link
      - ☑ High-speed Link 01
    - □ P2P(EIP)

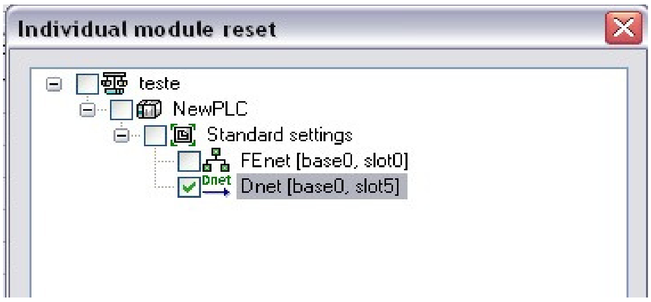
- Click on Online > Enable Link and check the option as shown in the screen below and then click on Write > Ok > Close:



Enable Link(HS Link,P2P)

- □ teste
  - □ NewPLC
    - □ High-speed Link
      - ☑ High-speed Link 01
      - □ High-speed Link 02
      - □ High-speed Link 03

- Click on Online > Reset > Reset Individual Module and select the option as shown in the screen below, then click on OK:



Individual module reset

- □ teste
  - □ NewPLC
    - □ Standard settings
      - □ FEnet [base0, slot0]
      - ☑ Dnet [base0, slot5]

And

- PLC; - Profibus communication module – XGL-PMEA; - Profibus device to be communicated with;

To run a program to test communication:

- On the device to be communicated with, set the device address;

- Find the GSD file of the device to be communicated with;

- Copy and paste the GSD file into the root of the Sycon software (C: > LG Industrial Systems > Sycon > Fildbus > Profibus > GSD);

- Connect a Female/Male RS232 cable between the PC and the Profibus Module (XGL_PMEA);

- Open the Sycon software, create a new Profibus file;

- Insert a master, which in this case will be the PLC;

- Select the COM-C-DPM > Add>> OK option;

- Click on Settings > Master Settings and if it is not checked, check the Buffered, host controlled option;

- Insert a slave below the master;

- Locate the GSD file in the list and add it;

- Click on Settings > Slave Configuration;

- Enter the address that was configured on the device to communicate in the Station Address field and in the Append Module button, add the inputs and outputs that
will be used;

- Close any program that is using the PC's COM port;

- Select the Mester0 device, go to Settings > Bus Parameter and select the communication speed according to the PC's communication speed;

- Click Online > Download;

- Click on the Connect COM1 button and check the COM 1 field: on the left side > Ok > Yes;

- The settings will be sent to the Profibus communication module;

Profibus Communication

—

- Create a new project in the XG5000 containing only one END in the program;
- In the XG5000, open the XG PD;
- Create a new file and connect;
- Click on Online > Read IO Information;
- Check the Slot number where the Pnet module is located
- Click on the High-speed Link tab > 2 clicks on High-speed Link 01;
- In Module type, select Pnet and in Slot Number, enter the number corresponding to the slot

observed previously.

- Click anywhere in the gray table on the right;
- Click on online > Sycon Upload;
- In the gray table on the right, enter a memory address in Read area (e.g.:

D0) and an address in Save area (e.g.: D50);

- Click on Online > Write Parameter;
- Click on Online > Reset > Reset Individual Module;
- Select only the Pnet module to be reset;
- Click on Online > Enable Link and check only High-speed Link 01 > Write > Ok > Close;

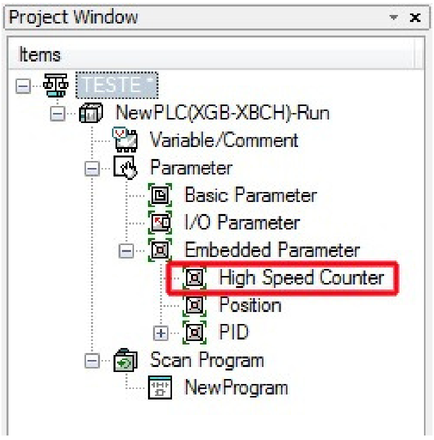
The communication is now configured, to test you can send data from the device to communicate, enter the XG5000 and in Monitor > Device Monitoring select memories D to check if the data is arriving in memory D50 onwards, which is the memory responsible for receiving data from the device to communicate as previously configured in the XG PD.

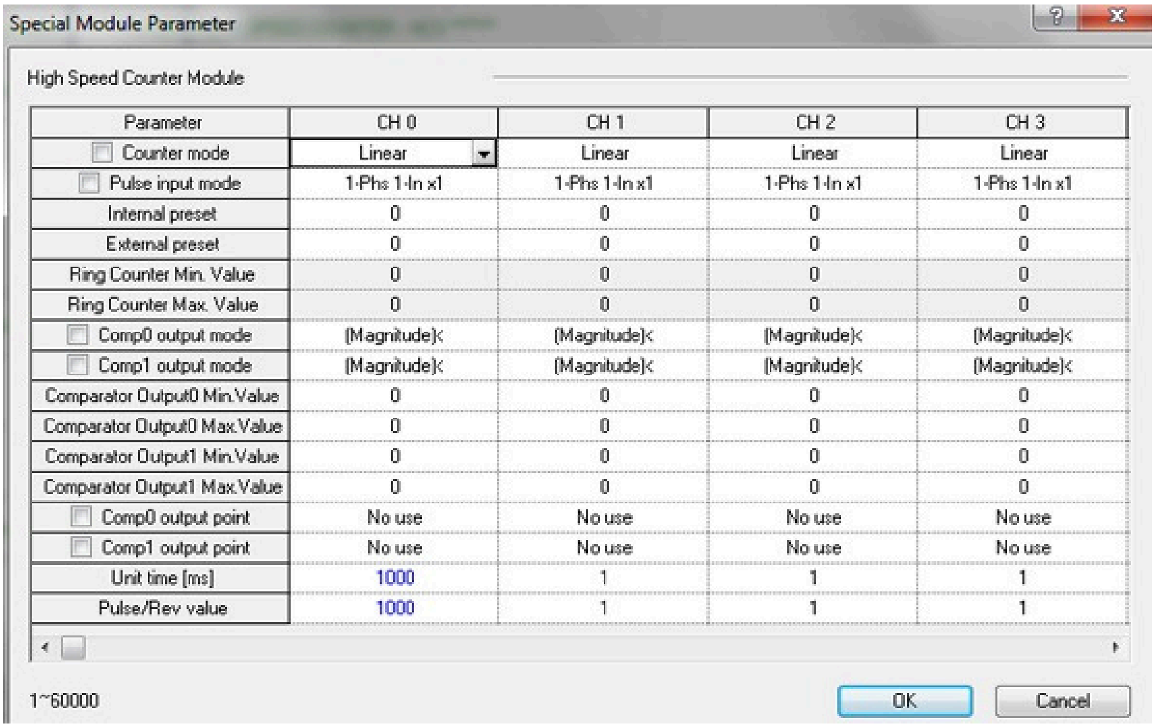
END.

The parameter configuration table for the HIGH SPEED COUNTER function is located in Embedded Parameter:

Encoder Configuration - HIGH SPEED COUNTER

On the XG5000:





High Speed Counter Module

Parameter	CH 0	CH 1	CH 2	CH 3
□ Counter mode	Linear	Linear	Linear	Linear
□ Pulse input mode	1-Phs 1-In x1	1-Phs 1-In x1	1-Phs 1-In x1	1-Phs 1-In x1
Internal preset	0	0	0	0
External preset	0	0	0	0
Ring Counter Min. Value	0	0	0	0
Ring Counter Max. Value	0	0	0	0
□ Comp0 output mode	(Magnitude)<	(Magnitude)<	(Magnitude)<	(Magnitude)<
□ Comp1 output mode	(Magnitude)<	(Magnitude)<	(Magnitude)<	(Magnitude)<
Comparator Output0 Min.Value	0	0	0	0
Comparator Output0 Max.Value	0	0	0	0
Comparator Output1 Min.Value	0	0	0	0
Comparator Output1 Max.Value	0	0	0	0
□ Comp0 output point	No use	No use	No use	No use
□ Comp1 output point	No use	No use	No use	No use
Unit time [ms]	1000	1	1	1
Pulse/Rev value	1000	1	1	1

1~60000