PLC AND SCADA LAB

Course Code: EIL66	**Credits:** 0:0:1
Pre – requisites: Process Instrumentation and Process Control	**Contact Hours:** 14
Course Coordinator: Dr. J V Alamelu and Dr. Nishi Shahnaj Haider	

List of Experiments:

1. Introduction to Rockwell Automation Software development environment, implementation of Basic Boolean functions.
2. PLC ladder logic & FBD programming to control the production line process using SIMBOX.
3. PLC ladder logic & FBD programming to control the rejection process using SIMBOX.
4. PLC ladder logic & FBD programming to control the batching process using SIMBOX and virtual lab (online simulation).
5. PLC ladder logic & FBD programming to control the pick and place process.
6. PLC ladder logic & FBD programming to control the traffic monitoring system using Virtual Lab (online simulation).
7. Introduction to Mitsubishi Software development environment, implementation of Basic Boolean functions.
8. PLC ladder logic programming to control the elevator using RS Logix and GX Works.
9. PLC ladder logic programming to control the bottle filling process using RS Logix and GX Works.
10. PLC ladder logic programming to control the coffee vending machine process using RS Logix and GX Works.
11. PLC ladder logic programming to configure Analog Input and output module to read and write current and voltage.
12. Configuring SCADA for I/O acquisition.
13. Ladder logic programming to control simple ON-OFF process using RTUs.
14. Ladder logic programming to control water level in overhead tank using RTUs.

Note: Follow up for the experiments will be announced in the laboratory during conduction. The questions will be based on logical modification of existing ladder diagrams/FBD/Virtual lab simulations (in online).

Text Books

1. Introduction to Programmable Logic Controllers by Garry Dunning, Thomson, 2nd edition, Thomson, ISBN: 981-240-625-5
2. Practical SCADA for Industry, David Bailey and Edwin Wright, An imprint of Elsevier, 2003, ISBN 07506 58053

Course Outcomes (COs):

The students should be able to:

1. Utilize Rockwell Automation Software and Mitsubishi Software for PLC programming. (PO-1,3,5,10,12, PSO-1,3)
2. Utilize Clear SCADA software for SCADA programming. (PO1,3,5,10,12, PSO-1,3)
3. Implement a prototype simulation plant process. (PO3,5,7,9,10, PSO-1,2,3)

Course Assessment and Evaluation:

Continuous Internal Evaluation CIE): 50 Marks		
Assessment tool	**Marks**	**Course outcomes attained**
Weekly evaluation of laboratory records/ reports after the conduction of every experiment. Follow up / Viva	20 10	CO1-CO3
Practical test	20	CO1-CO3
Semester End Examination (SEE)	**50**	CO1-CO3

Procedure for the creation of the Ladder Logic and Function Block Diagram in RSLogix 5000

→ Go to START
→ Open Rockwell Software
→ RsLogix5000

Creation of new project: (Ladder Logic):

→ Open new project
→ Vendor: Allen Bradley
→ Type: 1769_L32E Compact Logix 5332E controller
→ Revision: 20
→ Project Name: XYZ
→ OK
→ In controller organizer window, dropdown the main program and select main routine
→ In the controller organizer window, right click on Compact Bus Local
→ Select New Module
→ Select the model number as 1769-IQ6XOW4 ⟶ 6 point 24VDC sink/source input, 4 point AC/DC relay output
→ Select create, the new module window opens
→ Provide a name and select slot-1 (slot-2, if necessary), select close
→ In the editor window draw the ladder logic
→ Right click on the object and select new tag
→ Select type as ALIAS for actual I/Os or else select BASE
→ If type is ALIAS, then select ALIAS for: Local:1:I:Data0 to Data5 for inputs and Local:1:O:Data0 to Data3 for outputs
→ Save the program
→ Select communication and dropdown
→ Select Who Active
→ Select AB_ETH_1, Ethernet ⟶ 192.168.1.10 ⟶backplane
→ Configuration system ⟶ compact logix processor
→ Select download
→ Click on "NO Forces" option and select IO forcing
→ Select "Enable all IO forces"
→ Verify the code by providing FORCES ON and FORCES OFF options

Creation of New Project (Function Block Diagram)

- → Open new project
- → Vendor: Allen Bradley
- → Type: 1769_L32E Compact Logix 5332E controller
- → Revision: 20
- → Project Name: XYZ
- → OK
- → In controller organizer window, dropdown the main program and select main routine
- → In the controller organizer window, right click on Compact Bus Local
- → Select New Module
- → Select the model number as 1769-IQ6XOW4 → 6 point 24VDC sink/source input, 4-point AC/DC relay output
- → Select create, the new module window opens
- → Provide a name and select slot-1 (slot-2, if necessary), select close
- → In the editor window draw the ladder logic for jump to sub-routine instruction (JSR)
- → In the controller organizer window, right click on main program
- → Select new routine and provide a name
- → Same routine name has to be provided in the JSR instruction
- → In the editor window of the function block diagram draw the required blocks and make the suitable connections
- → Right click on the object and select new tag
- → Select type as ALIAS for actual I/Os or else select BASE
- → If type is ALIAS, then select ALIAS for: Local:1:I:Data0 to Data5 for inputs and Local:1:O:Data0 to Data3 for outputs
- → Save the program
- → Select communication and dropdown
- → Select Who Active
- → Select AB_ETH_1, Ethernet → 192.168.1.10 →backplane
- → Configuration system → compact logix processor
- → Select download
- → Click on "NO Forces" option and select IO forcing
- → Select "Enable all IO forces"
- → Verify the code by providing FORCES ON and FORCES OFF options

Experiment - 1

BASIC GATES

Objective:

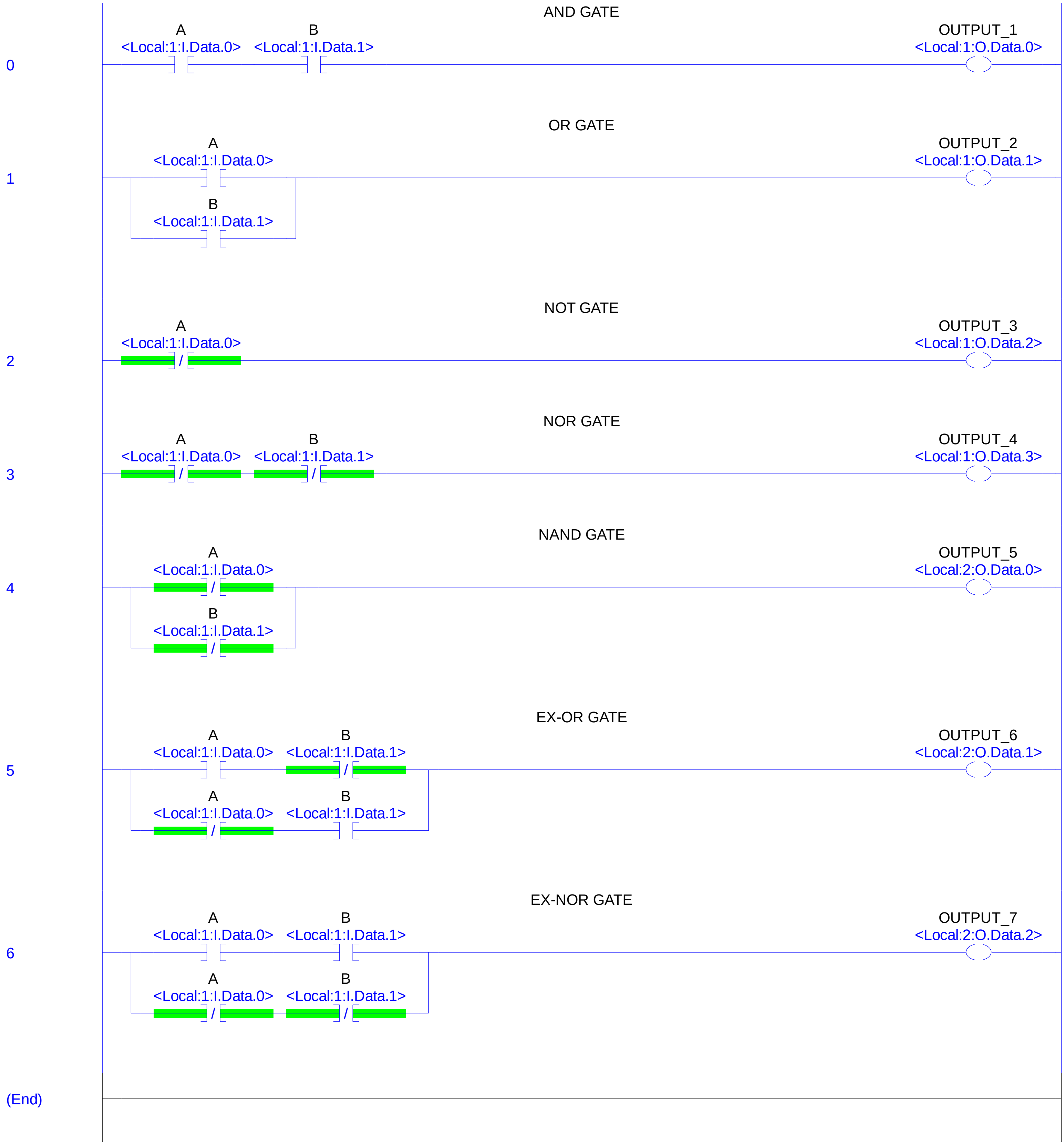
To verify the working of basic gates with their truth-table using ladder diagram in RSLogix5000.

Procedure:

- To develop the basic gates logic using normally open, normally closed and output coil instructions in the ladder diagram.
- Include the rung comments and rung wise explanation.

Inference: Performed the basic logic gate operation using PLC.

Outcomes:


AND GATE
0
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_1
<Local:1:O.Data.0>
OR GATE
1
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_2
<Local:1:O.Data.1>
NOT GATE
2
A
<Local:1:I.Data.0>
OUTPUT_3
<Local:1:O.Data.2>
NOR GATE
3
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_4
<Local:1:O.Data.3>
NAND GATE
4
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_5
<Local:2:O.Data.0>
EX-OR GATE
5
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_6
<Local:2:O.Data.1>
EX-NOR GATE
6
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
A
<Local:1:I.Data.0>
B
<Local:1:I.Data.1>
OUTPUT_7
<Local:2:O.Data.2>
(End)

Experiment - 2

MULTIPLEXER, DE-MULTIPLEXER and DECODER

Objective:

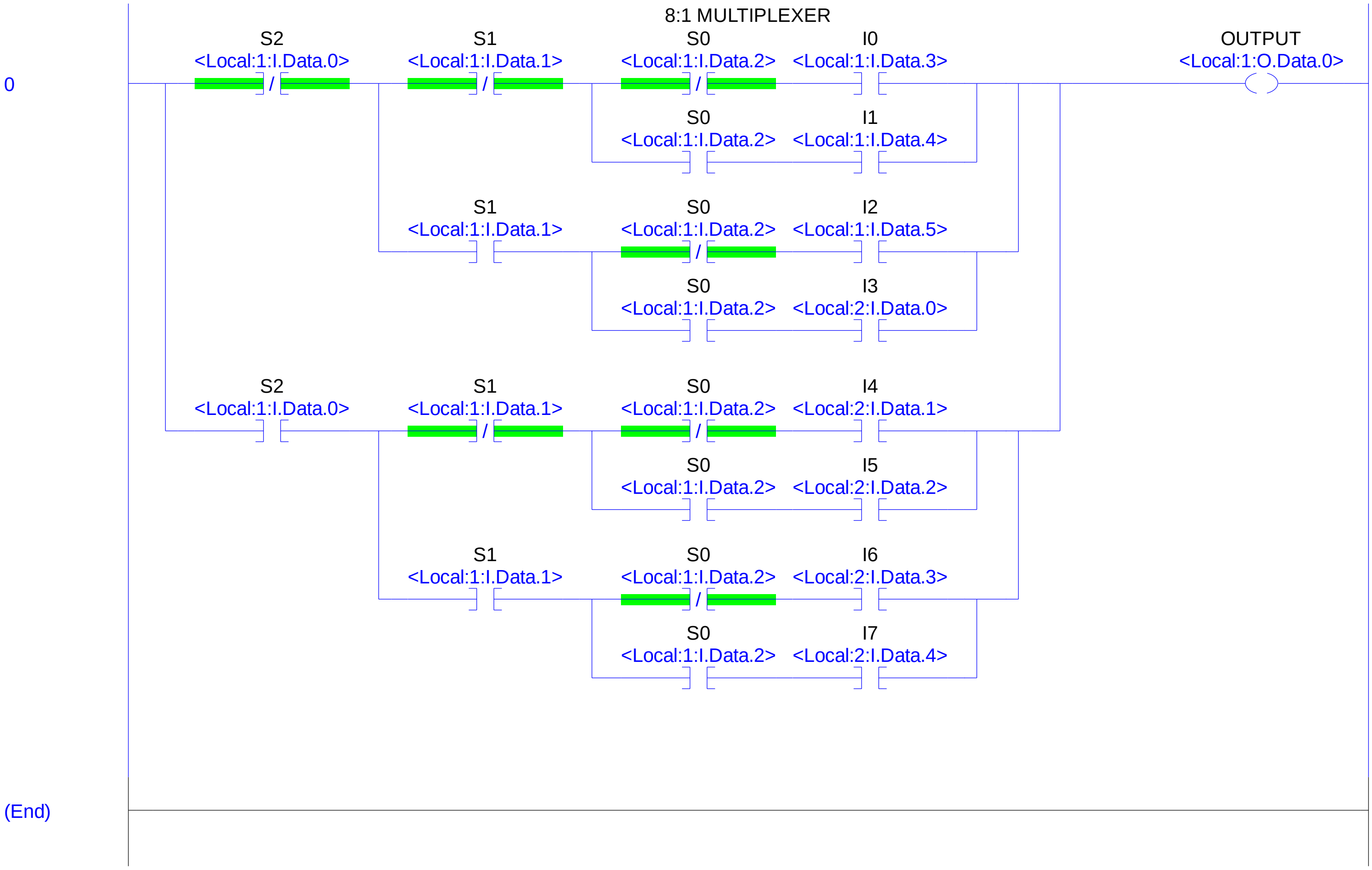
To verify the working of 8:1 Multiplexer, 1:8 De-multiplexers and 3:8 Decoder with their truth-table using ladder diagram in RSLogix5000.

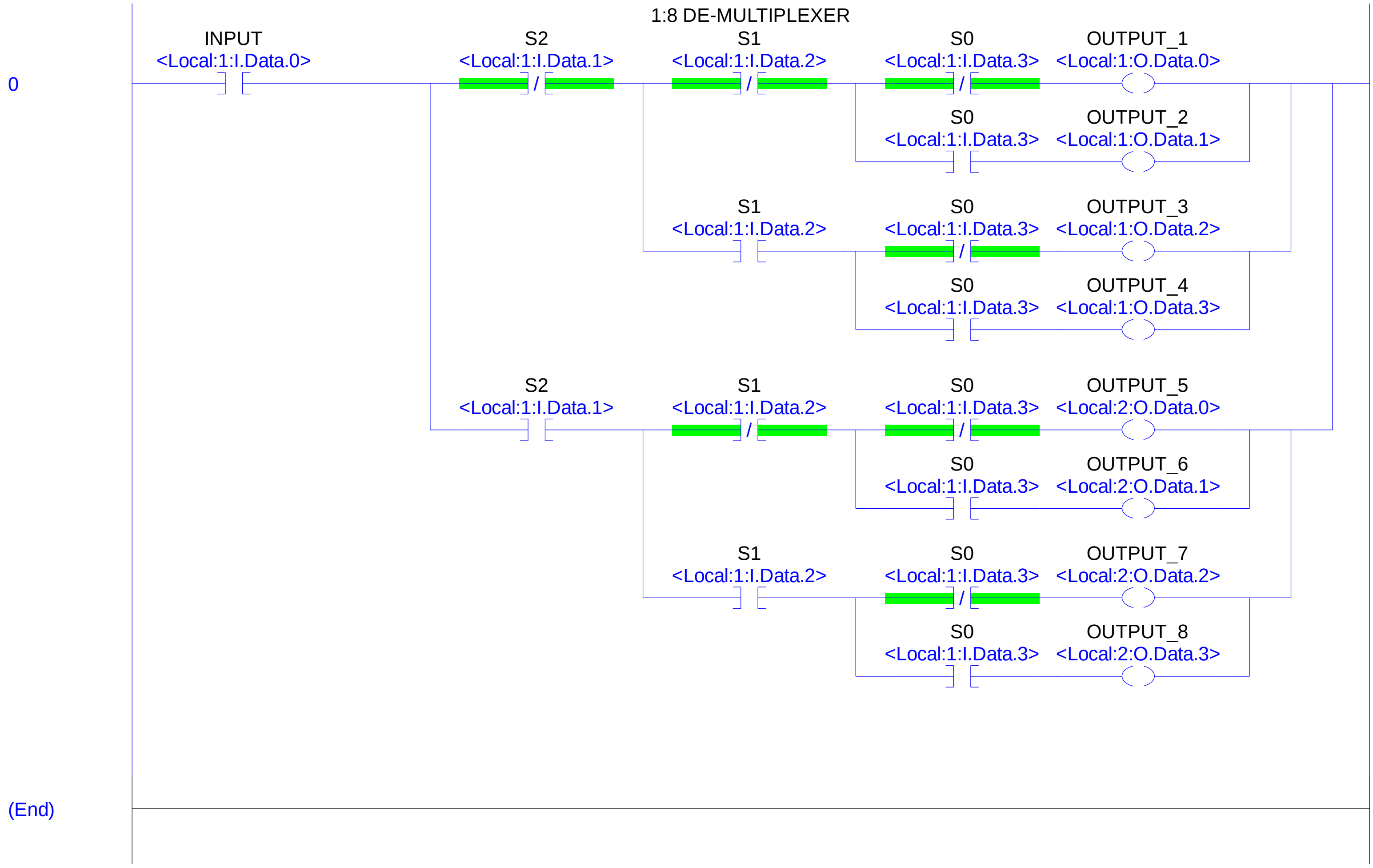
Procedure:

- To develop the 8:1 Multiplexer, 1:8 De-multiplexer and 3:8 Decoder logic using normally open, normally closed and output coil instructions in the ladder diagram.
- Include the rung comments and rung-wise explanation.

Follow up: The ladder diagram should be executed using Alias and Simbox

Inference: Designed the 8:1 Multiplexer, 1:8 De-multiplexer and 3:8 Decoder using PLC Ladder Logic. Also, verified their functioning with respect to the truth-table.


8:1 MULTIPLEXER
0
S2
<Local:1:I.Data.0>
S1
<Local:1:I.Data.1>
S0
<Local:1:I.Data.2>
I0
<Local:1:I.Data.3>
OUTPUT
<Local:1:O.Data.0>
S0
<Local:1:I.Data.2>
I1
<Local:1:I.Data.4>
S1
<Local:1:I.Data.1>
S0
<Local:1:I.Data.2>
I2
<Local:1:I.Data.5>
S0
<Local:1:I.Data.2>
I3
<Local:2:I.Data.0>
S2
<Local:1:I.Data.0>
S1
<Local:1:I.Data.1>
S0
<Local:1:I.Data.2>
I4
<Local:2:I.Data.1>
S0
<Local:1:I.Data.2>
I5
<Local:2:I.Data.2>
S1
<Local:1:I.Data.1>
S0
<Local:1:I.Data.2>
I6
<Local:2:I.Data.3>
S0
<Local:1:I.Data.2>
I7
<Local:2:I.Data.4>
(End)


1:8 DE-MULTIPLEXER
0
INPUT
<Local:1:I.Data.0>
S2
<Local:1:I.Data.1>
S1
<Local:1:I.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_1
<Local:1:O.Data.0>
S0
<Local:1:I.Data.3>
OUTPUT_2
<Local:1:O.Data.1>
S1
<Local:1:I.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_3
<Local:1:O.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_4
<Local:1:O.Data.3>
S2
<Local:1:I.Data.1>
S1
<Local:1:I.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_5
<Local:2:O.Data.0>
S0
<Local:1:I.Data.3>
OUTPUT_6
<Local:2:O.Data.1>
S1
<Local:1:I.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_7
<Local:2:O.Data.2>
S0
<Local:1:I.Data.3>
OUTPUT_8
<Local:2:O.Data.3>
(End)

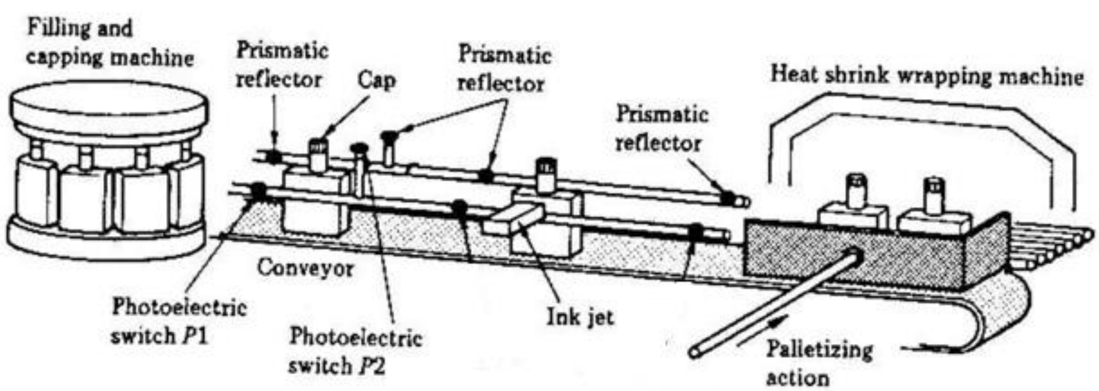
Experiment - 3

PRODUCTION LINE PROCESS

Objective:
To implement the production line process using ladder diagram and function block diagram in RSLogix5000.

Problem Statement:
The PLC task is to organize the production line process. Cans are filled with fluid and capped before passing into a conveyor. The photoelectric switch P1 is used to sense the filling process and P2 is used to check that each can has a cap. After filling and capping the label process has to be carried. A required number of bottles have to be wrapped and the process has to continue. Utilize the start-stop logic.

Process Diagram:

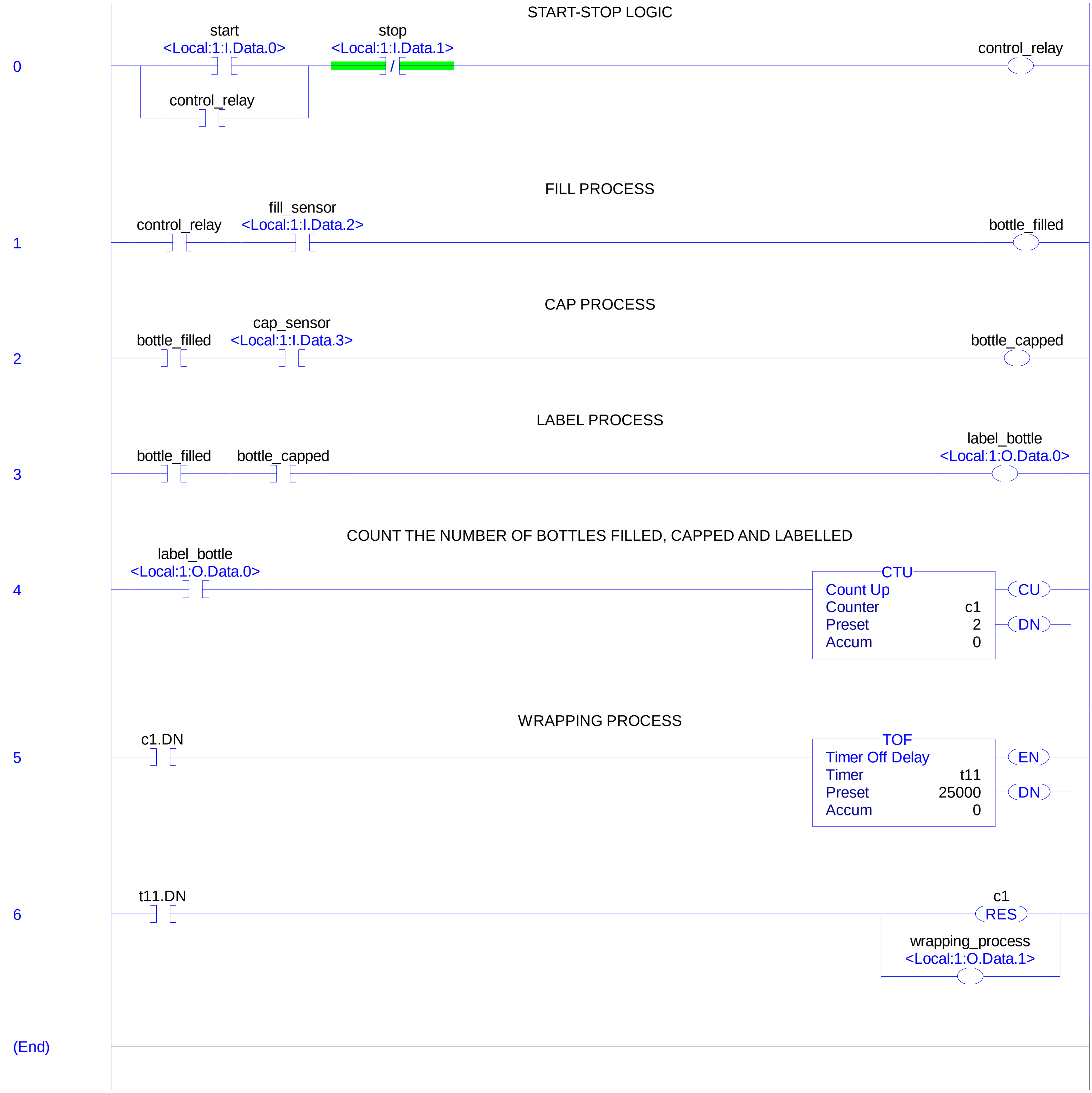


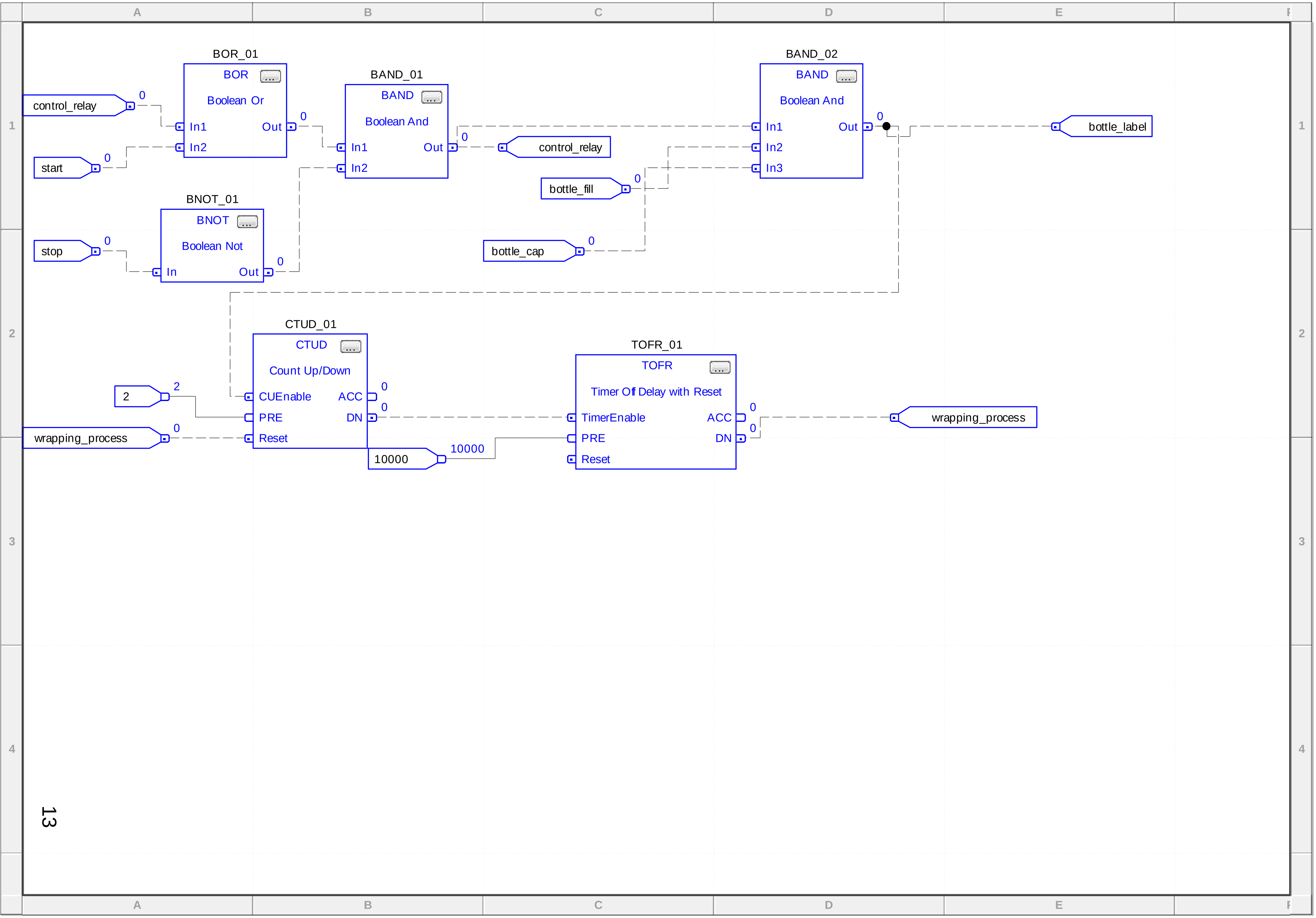
Procedure:

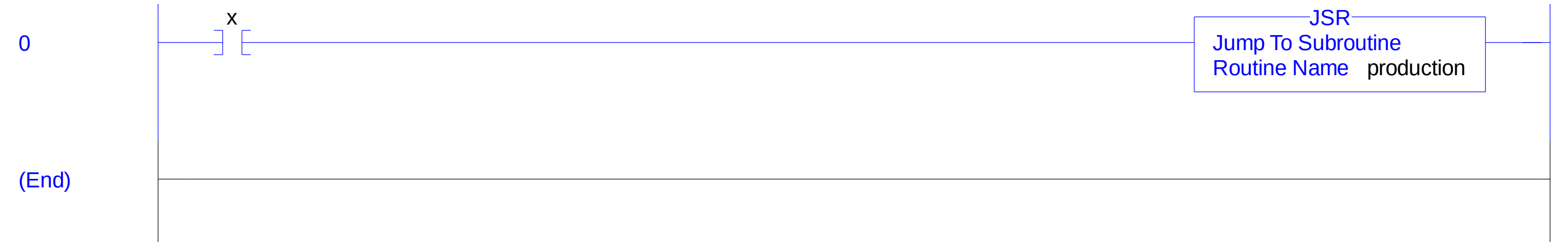
- Develop the logic using normally open, normally closed, output coil, timer and counter instructions in the ladder diagram.
- Include the rung comments and rung wise explanation for LD.
- Develop the logic using Boolean AND, Boolean OR, Boolean NOT, counter, timer instructions in the function block diagram.
- Include the block wise explanation for FBD.

Follow up:

Inference: Developed ladder logic and functional block diagram to automate the production line process.


START-STOP LOGIC
0
start
<Local:1:I.Data.0>
stop
<Local:1:I.Data.1>
control_relay
control_relay
FILL PROCESS
1
control_relay
fill_sensor
<Local:1:I.Data.2>
bottle_filled
CAP PROCESS
2
bottle_filled
cap_sensor
<Local:1:I.Data.3>
bottle_capped
LABEL PROCESS
3
bottle_filled
bottle_capped
label_bottle
<Local:1:O.Data.0>
COUNT THE NUMBER OF BOTTLES FILLED, CAPPED AND LABELLED
4
label_bottle
<Local:1:O.Data.0>
CTU
Count Up
Counter c1
Preset 2
Accum 0
CU
DN
WRAPPING PROCESS
5
c1.DN
TOF
Timer Off Delay
Timer t11
Preset 25000
Accum 0
EN
DN
6
t11.DN
c1
RES
wrapping_process
<Local:1:O.Data.1>
(End)


A
B
C
D
E
1
2
3
4
BOR_01
BOR
Boolean Or
In1
In2
Out
BAND_01
BAND
Boolean And
In1
In2
Out
BAND_02
BAND
Boolean And
In1
In2
In3
Out
BNOT_01
BNOT
Boolean Not
In
Out
CTUD_01
CTUD
Count Up/Down
CUEnable
PRE
Reset
ACC
DN
TOFR_01
TOFR
Timer Off Delay with Reset
TimerEnable
PRE
Reset
ACC
DN
control_relay
start
stop
bottle_fill
bottle_cap
bottle_label
wrapping_process
2
10000
0


0
x
JSR
Jump To Subroutine
Routine Name production
(End)

Experiment - 4

PICK and PLACE PROCESS

Objective:

To implement the pick and place process using ladder diagram and function block diagram in RSLogix5000.

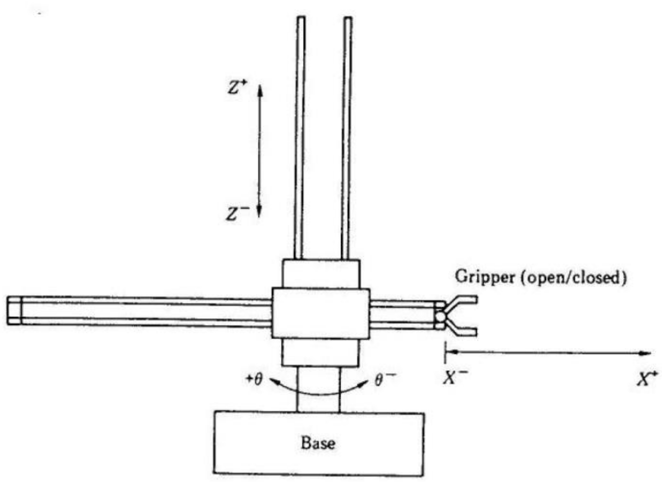
Problem Statement:

The PLC task is to:

a) Move the gripper to X+ position
b) Close the gripper so that it takes hold of a component
c) Rotate the gripper through 180^0 to the θ+ position
d) Release the component
e) Rotate the gripper back to the θ- position so that the pick and place operation may be repeated.

Utilize the start-stop logic.

Process Diagram:

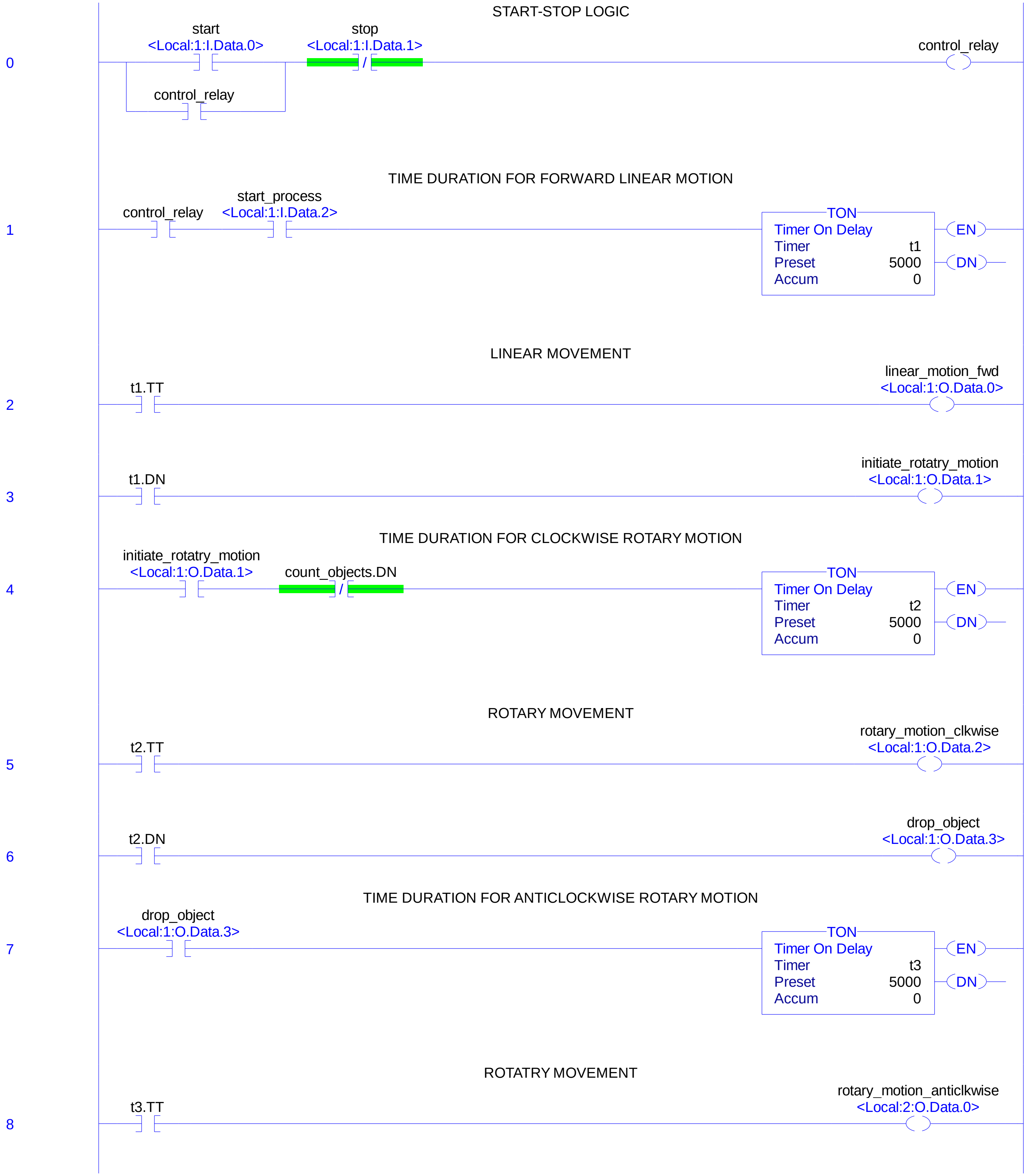


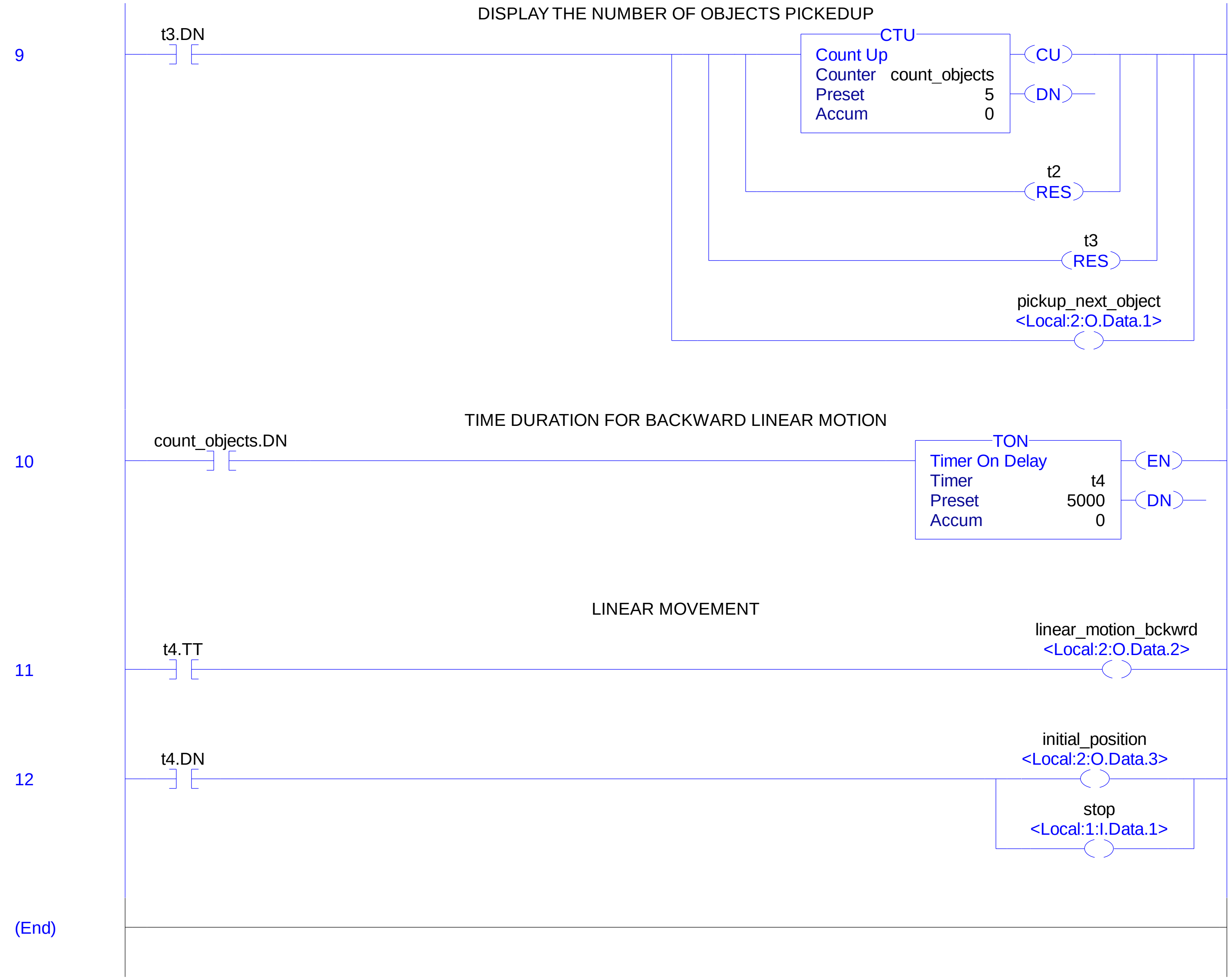
Procedure:

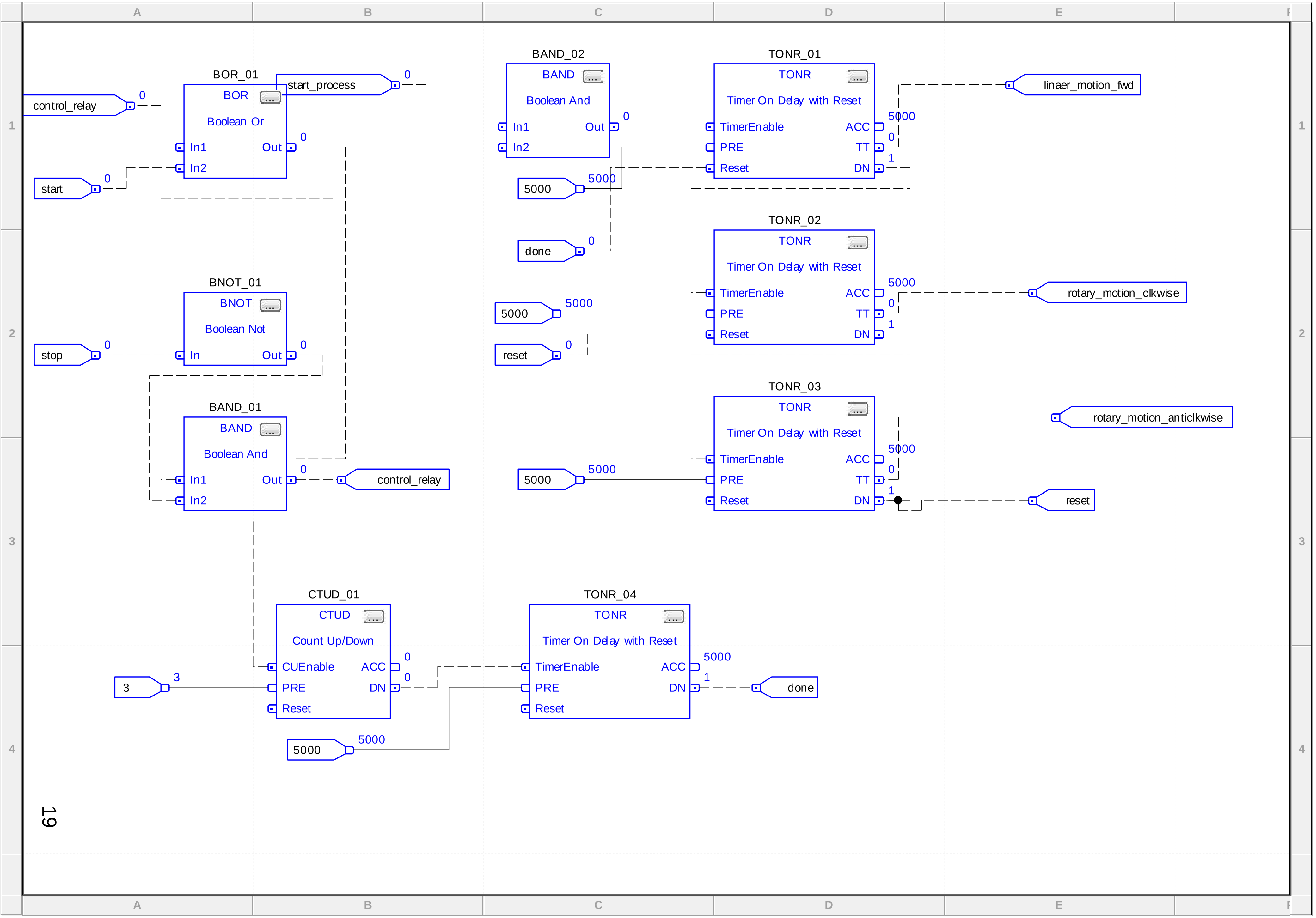
- Develop the logic using normally open, normally closed, output coil and timer instructions in the ladder diagram.
- Include the rung comments and rung wise explanation for LD.
- Develop the logic using Boolean AND, Boolean OR, Boolean NOT and timer instructions in the function block diagram.
- Include the block wise explanation for FBD.

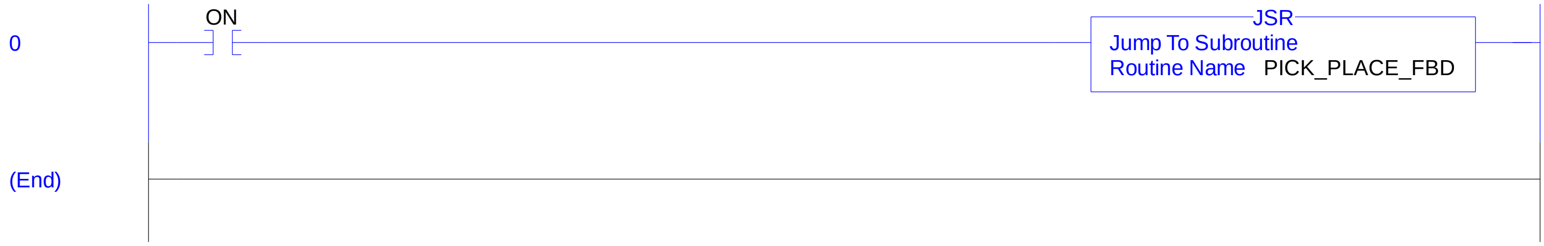
Follow up:

Inference: Implemented the PLC ladder diagram and functional block diagram for handling the pick and place process.


START-STOP LOGIC
0
start
<Local:1:I.Data.0>
stop
<Local:1:I.Data.1>
control_relay
control_relay
TIME DURATION FOR FORWARD LINEAR MOTION
1
control_relay
start_process
<Local:1:I.Data.2>
TON
Timer On Delay
Timer t1
Preset 5000
Accum 0
EN
DN
LINEAR MOVEMENT
2
t1.TT
linear_motion_fwd
<Local:1:O.Data.0>
3
t1.DN
initiate_rotatry_motion
<Local:1:O.Data.1>
TIME DURATION FOR CLOCKWISE ROTARY MOTION
4
initiate_rotatry_motion
<Local:1:O.Data.1>
count_objects.DN
TON
Timer On Delay
Timer t2
Preset 5000
Accum 0
EN
DN
ROTARY MOVEMENT
5
t2.TT
rotary_motion_clkwise
<Local:1:O.Data.2>
6
t2.DN
drop_object
<Local:1:O.Data.3>
TIME DURATION FOR ANTICLOCKWISE ROTARY MOTION
7
drop_object
<Local:1:O.Data.3>
TON
Timer On Delay
Timer t3
Preset 5000
Accum 0
EN
DN
ROTATRY MOVEMENT
8
t3.TT
rotary_motion_anticlkwise
<Local:2:O.Data.0>


DISPLAY THE NUMBER OF OBJECTS PICKEDUP
9
t3.DN
CTU
Count Up
Counter count_objects
Preset 5
Accum 0
CU
DN
t2
RES
t3
RES
pickup_next_object
<Local:2:O.Data.1>
TIME DURATION FOR BACKWARD LINEAR MOTION
10
count_objects.DN
TON
Timer On Delay
Timer t4
Preset 5000
Accum 0
EN
DN
LINEAR MOVEMENT
11
t4.TT
linear_motion_bckwrd
<Local:2:O.Data.2>
12
t4.DN
initial_position
<Local:2:O.Data.3>
stop
<Local:1:I.Data.1>
(End)


BOR_01
BOR
Boolean Or
In1
In2
Out
BNOT_01
BNOT
Boolean Not
In
Out
BAND_01
BAND
Boolean And
In1
In2
Out
BAND_02
BAND
Boolean And
In1
In2
Out
TONR_01
TONR_02
TONR_03
TONR_04
TONR
Timer On Delay with Reset
TimerEnable
PRE
Reset
ACC
TT
DN
CTUD_01
CTUD
Count Up/Down
CUEnable
PRE
Reset
ACC
DN
control_relay
start
stop
start_process
done
reset
linaer_motion_fwd
rotary_motion_clkwise
rotary_motion_anticlkwise
5000
3
0
1
19


0
ON
JSR
Jump To Subroutine
Routine Name PICK_PLACE_FBD
(End)

Experiment - 5

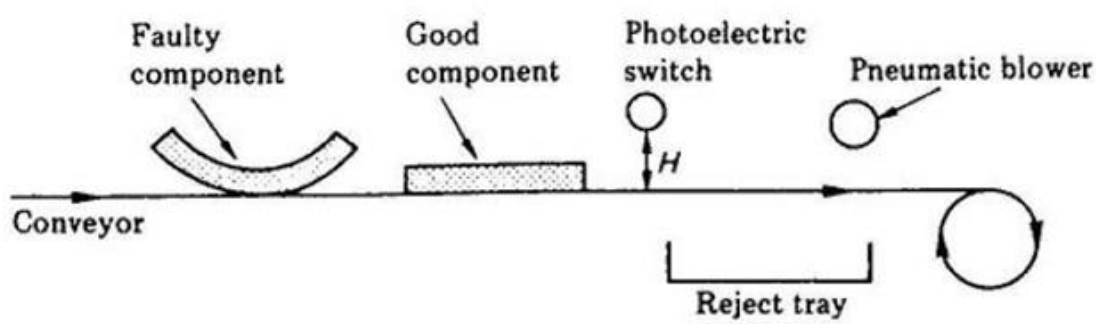
REJECTION LINE PROCESS

Objective:

To implement the rejection line process using ladder diagram and function block diagram in RSLogix5000.

Problem Statement:
The PLC task is to detect and reject faulty components. Components are transported on a conveyor past a retro-reflective type photoelectric switch. The photoelectric switch is positioned at a height (H) above the conveyor where (H) represents a tolerance value for component height. Good components pass underneath the photoelectric switch and no signal is generated. Faulty components break the light beam twice as they pass the photoelectric switch. Utilize the start-stop logic.

Process Diagram:

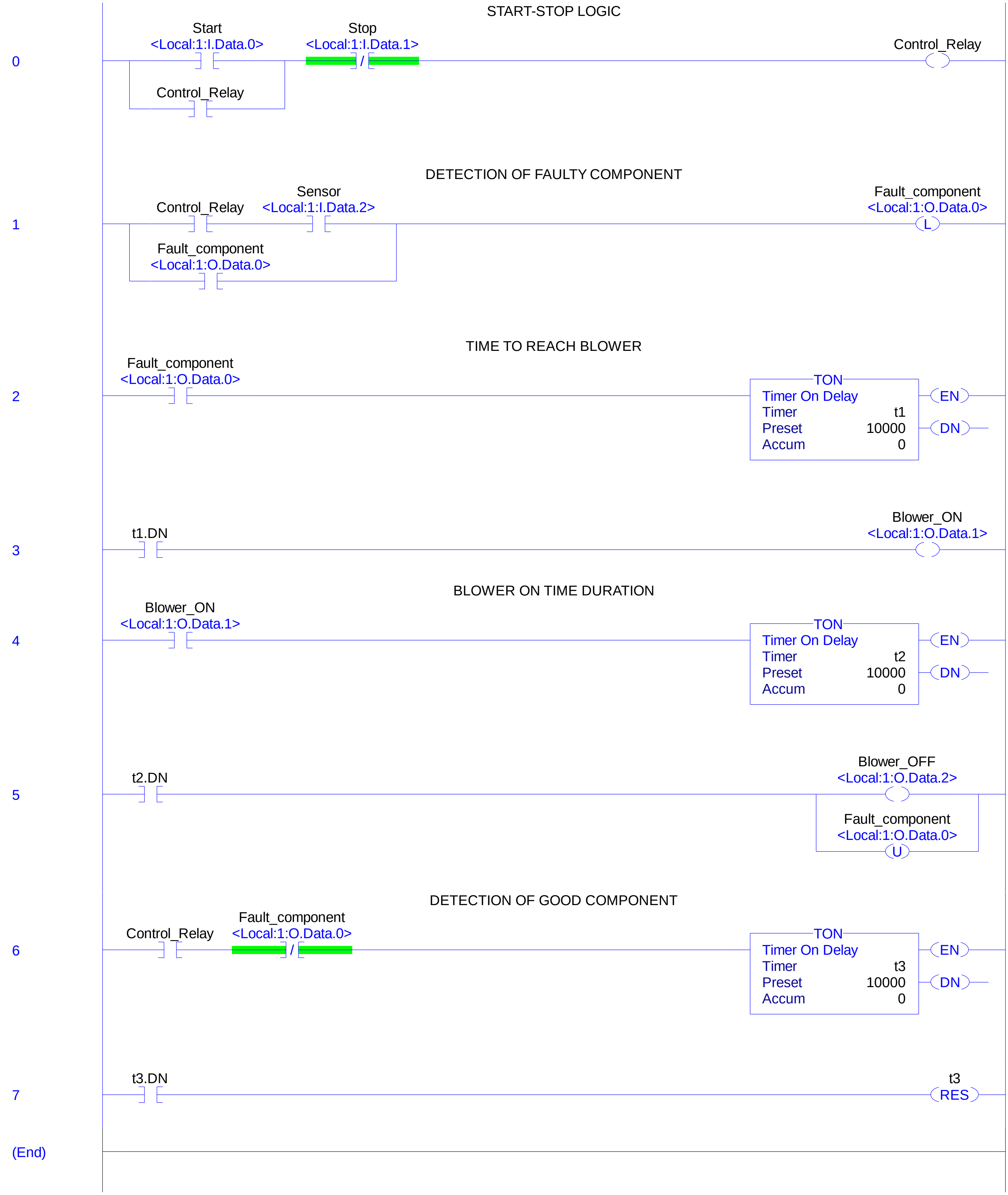


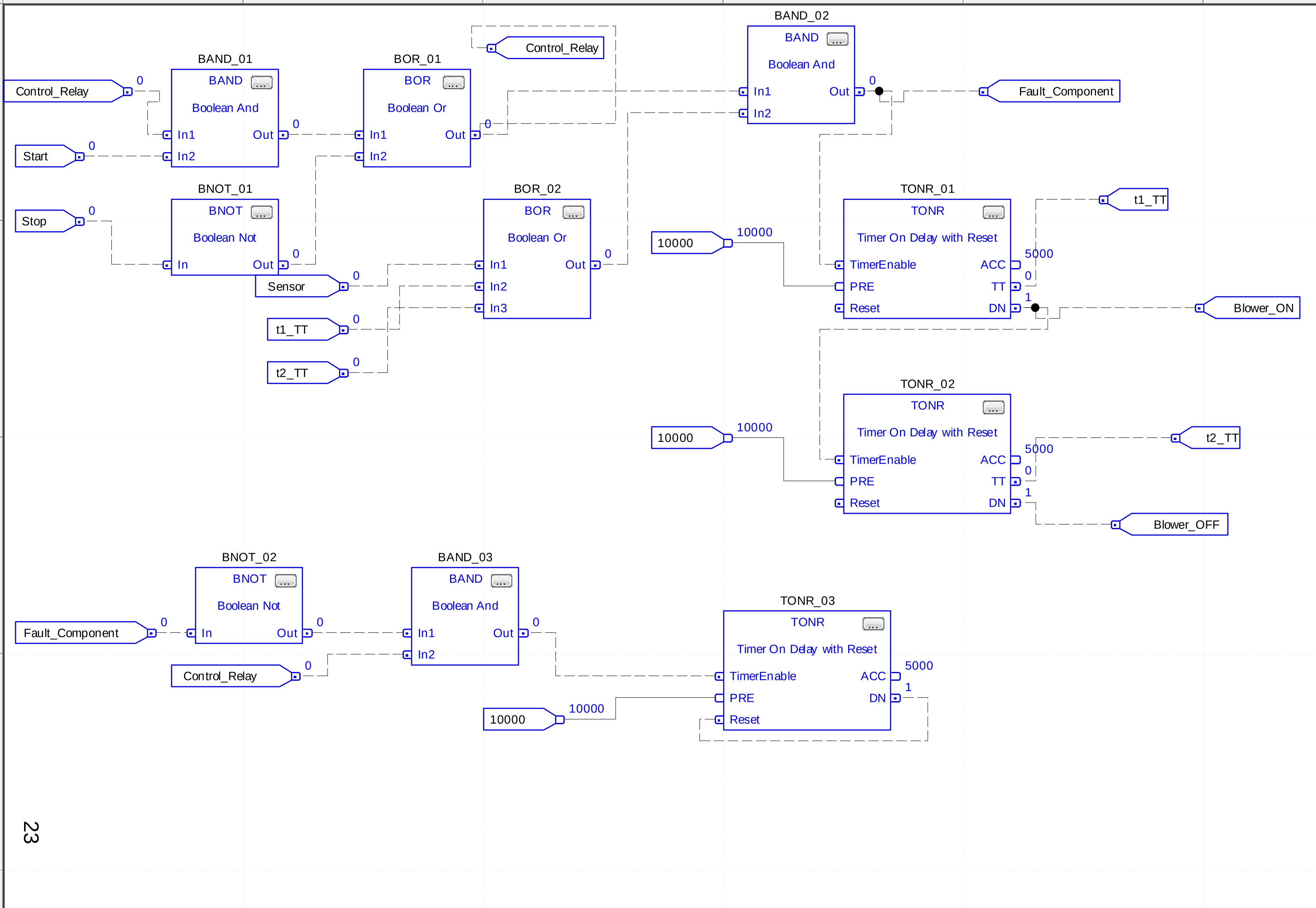
Procedure:

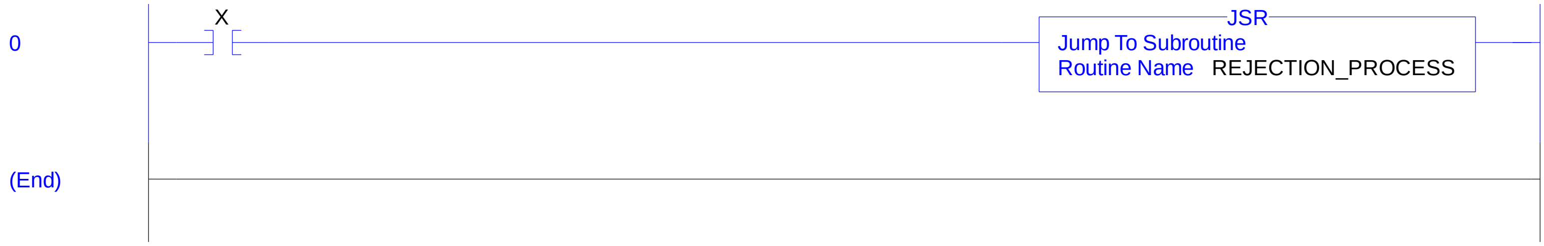
- Develop the logic using normally open, normally closed, output coil and timer instructions in the ladder diagram.
- Include the rung comments and rung wise explanation for LD.
- Develop the logic using Boolean AND, Boolean OR, Boolean NOT and timer instructions in the function block diagram.
- Include the block wise explanation for FBD.

Follow up:

Inference: Rejection line process is successfully implemented using PLC ladder diagram as well as with functional block diagram.


START-STOP LOGIC
0
Start
<Local:1:I.Data.0>
Stop
<Local:1:I.Data.1>
Control_Relay
Control_Relay
DETECTION OF FAULTY COMPONENT
1
Control_Relay
Sensor
<Local:1:I.Data.2>
Fault_component
<Local:1:O.Data.0>
L
Fault_component
<Local:1:O.Data.0>
TIME TO REACH BLOWER
2
Fault_component
<Local:1:O.Data.0>
TON
Timer On Delay
Timer t1
Preset 10000
Accum 0
EN
DN
3
t1.DN
Blower_ON
<Local:1:O.Data.1>
BLOWER ON TIME DURATION
4
Blower_ON
<Local:1:O.Data.1>
TON
Timer On Delay
Timer t2
Preset 10000
Accum 0
EN
DN
5
t2.DN
Blower_OFF
<Local:1:O.Data.2>
Fault_component
<Local:1:O.Data.0>
U
DETECTION OF GOOD COMPONENT
6
Control_Relay
Fault_component
<Local:1:O.Data.0>
TON
Timer On Delay
Timer t3
Preset 10000
Accum 0
EN
DN
7
t3.DN
t3
RES
(End)


BAND_01
BAND
Boolean And
In1
In2
Out
BOR_01
BOR
Boolean Or
BAND_02
BNOT_01
BNOT
Boolean Not
In
BOR_02
In3
TONR_01
TONR
Timer On Delay with Reset
TimerEnable
PRE
Reset
ACC
TT
DN
TONR_02
TONR_03
BNOT_02
BAND_03
Control_Relay
Start
Stop
Sensor
t1_TT
t2_TT
Fault_Component
Blower_ON
Blower_OFF
10000
5000
0
1


0
X
JSR
Jump To Subroutine
Routine Name REJECTION_PROCESS
(End)

Experiment - 6

BATCH PROCESS

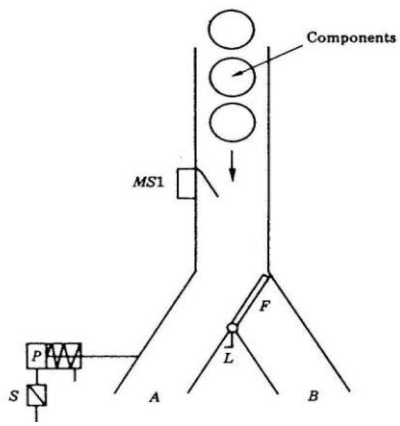
Objective:

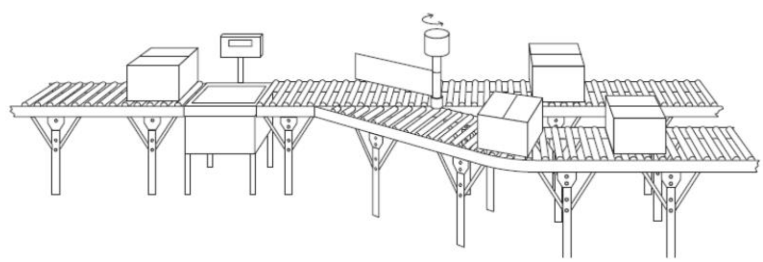
To implement the batch process using ladder diagram and function block diagram in RSLogix5000.

Problem Statement:

The PLC task is to control a simple machine which counts and batches components moving along a conveyor. Utilize the start-stop logic.

Process Diagram:





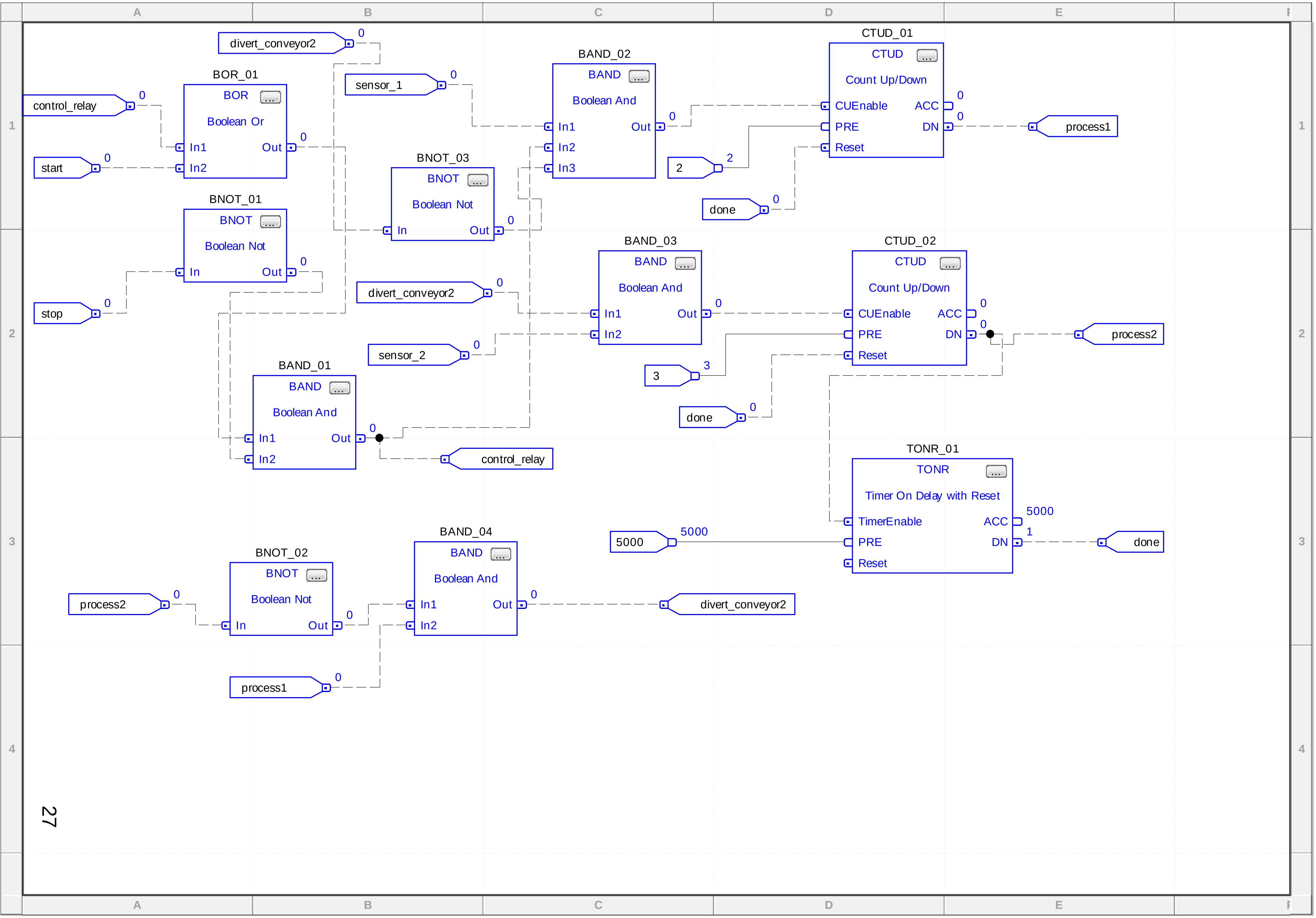
Procedure:

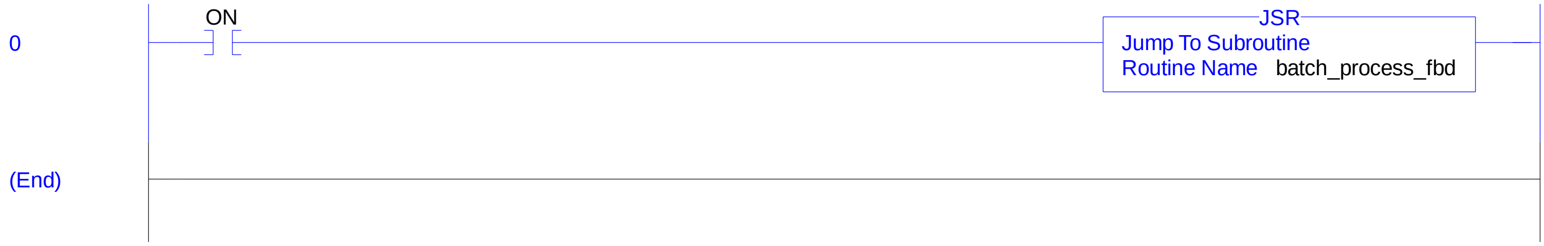
- Develop the logic using normally open, normally closed, output coil, counter and timer instructions in the ladder diagram.
- Include the rung comments and rung wise explanation for LD.
- Develop the logic using Boolean AND, Boolean OR, Boolean NOT, counter and timer instructions in the function block diagram.
- Include the block wise explanation for FBD.

Inferences

Developed the ladder logic to demonstrate the batch process using the PLC ladder logic.


START-STOP LOGIC
0
start
<Local:1:I.Data.0>
stop
<Local:1:I.Data.1>
control_relay
control_relay
PRODUCTS MOVING TOWARDS CONVEYOR-1
1
control_relay
sensor_1
<Local:1:I.Data.2>
divert_torwards_conveyor_2
<Local:1:O.Data.0>
CTU
Count Up
Counter counter_1
Preset 2
Accum 0
CU
DN
2
counter_1.DN
divert_torwards_conveyor_2
<Local:1:O.Data.0>
L
PRODUCTS MOVING TOWARDS CONVEYOR-2
3
sensor_2
<Local:1:I.Data.3>
divert_torwards_conveyor_2
<Local:1:O.Data.0>
CTU
Count Up
Counter counter_2
Preset 3
Accum 0
CU
DN
4
counter_2.DN
divert_towards_conveyor_2
<Local:1:O.Data.0>
U
process_complete
5
process_complete
TON
Timer On Delay
Timer delay
Preset 5000
Accum 0
EN
DN
RESET OF COUNTER VALUE TO CARRY OUT THE NEXT CONSECUTIVE PROCESS
6
delay.DN
counter_1
RES
counter_2
RES
(End)


divert_conveyor2
control_relay
start
BOR_01
BOR
Boolean Or
In1
In2
Out
sensor_1
BNOT_03
BNOT
Boolean Not
In
Out
BNOT_01
BNOT
Boolean Not
In
Out
stop
BAND_02
BAND
Boolean And
In1
In2
In3
Out
CTUD_01
CTUD
Count Up/Down
CUEnable
PRE
Reset
ACC
DN
process1
2
done
BAND_03
BAND
Boolean And
In1
In2
Out
divert_conveyor2
sensor_2
CTUD_02
CTUD
Count Up/Down
CUEnable
PRE
Reset
ACC
DN
process2
3
done
BAND_01
BAND
Boolean And
In1
In2
Out
control_relay
TONR_01
TONR
Timer On Delay with Reset
TimerEnable
PRE
Reset
ACC
DN
5000
done
BNOT_02
BNOT
Boolean Not
In
Out
process2
BAND_04
BAND
Boolean And
In1
In2
Out
divert_conveyor2
process1


0
ON
JSR
Jump To Subroutine
Routine Name batch_process_fbd
(End)

Experiment - 7

BOTTLE FILLING PROCESS

Objective:

To implement the bottle filling process using ladder diagram in RSLogix5000 and GX Works.

Problem Statement:

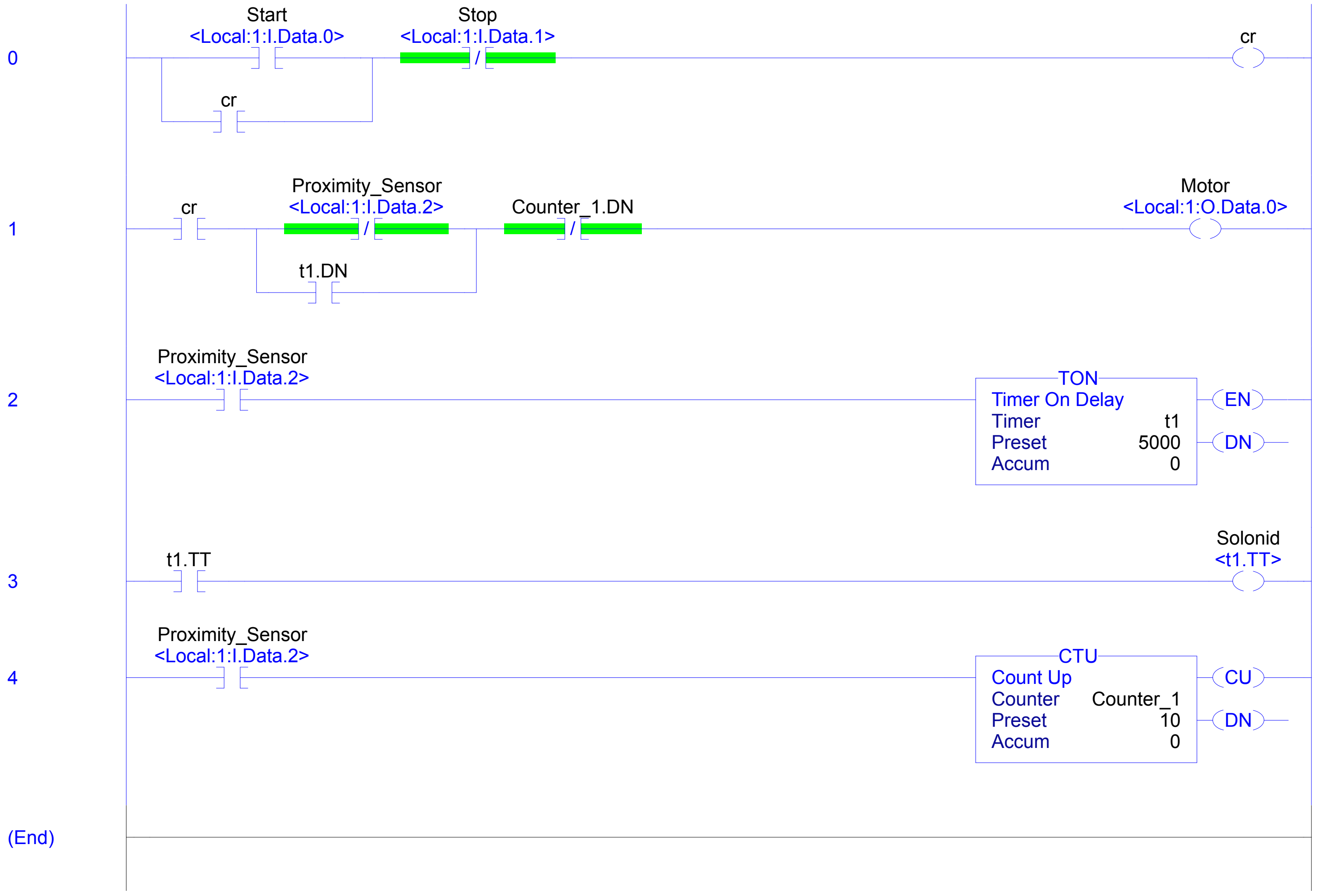
The PLC task is to control the bottle filling process. The model has a proximity sensor to presence the absence or absence of the bottle. When the bottle is present the solenoid valve should be in the ON condition and fluid should be filled. After the bottle is filled the DC motor is activated to move the conveyor belt to place the next bottle. Count the number of bottles also. Utilize the start-stop logic.

Procedure:

- To develop the bottle filling process using normally open, normally closed, output coil, timer and counter instructions in the ladder diagram.
- Include the rung comments and rung wise explanation.

Follow up:

Inference: The developed PLC ladder diagram is successfully controlling the bottle filling process.


0
Start
<Local:1:I.Data.0>
Stop
<Local:1:I.Data.1>
cr
cr
1
cr
Proximity_Sensor
<Local:1:I.Data.2>
Counter_1.DN
Motor
<Local:1:O.Data.0>
t1.DN
2
Proximity_Sensor
<Local:1:I.Data.2>
TON
Timer On Delay
Timer t1
Preset 5000
Accum 0
EN
DN
3
t1.TT
Solonid
<t1.TT>
4
Proximity_Sensor
<Local:1:I.Data.2>
CTU
Count Up
Counter Counter_1
Preset 10
Accum 0
CU
DN
(End)

Experiment - 8

ELEVATOR PROCESS

Objective:

To implement the elevator process using ladder diagram in RSLogix5000 and GX Works.

Problem Statement:

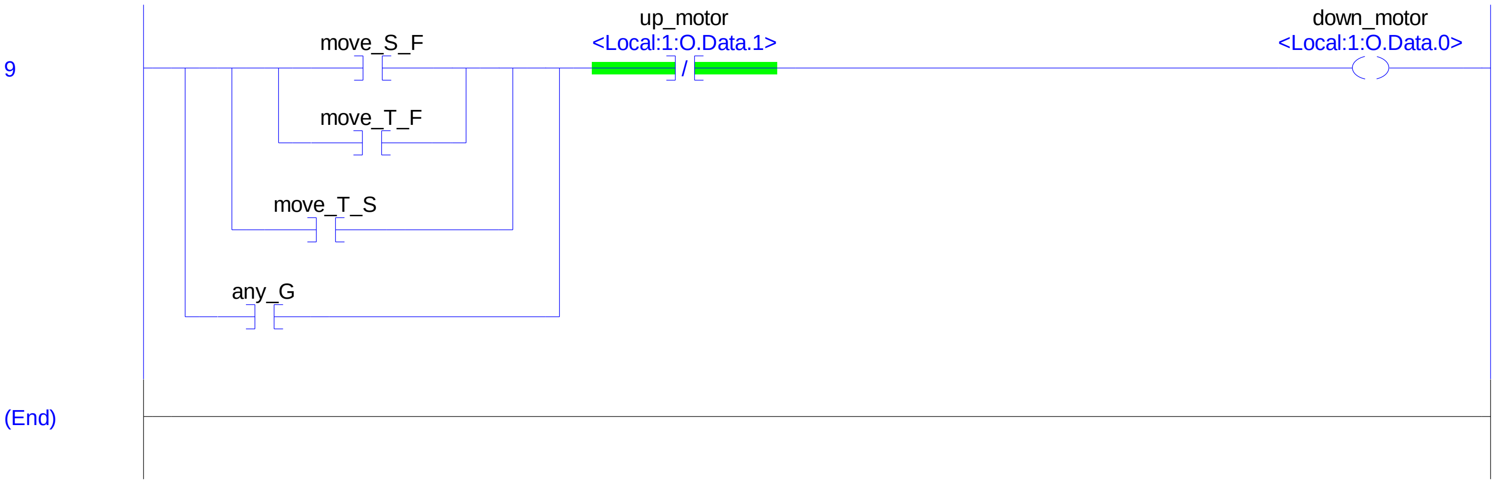
The PLC task is to control the elevator process. This is a three-level elevator process model. There are inputs to select the floor to which the elevator should move, when pressed a high level is fed to the corresponding input. There are limit switches which give the information regarding the carriage reaching a particular level; when a particular level is reached the proximity sensor gives a high signal. The motor is a bi-directional DC motor; there are relays which can be used to switch direction.

Procedure:

- To develop the elevator process using normally open, normally closed and output coil instructions in the ladder diagram.
- Include the rung comments and rung wise explanation.

Inference: Implemented the required PLC ladder diagram to automate the elevator process.


0
select_G
<Local:1:I.Data.0>
sensor_G
<Local:2:I.Data.0>
any_G
any_G
1
select_T
<Local:1:I.Data.3>
sensor_T
<Local:2:I.Data.3>
any_T
any_T
2
select_F
<Local:1:I.Data.1>
sensor_G
<Local:2:I.Data.0>
sensor_F
<Local:2:I.Data.1>
move_G_F
move_G_F
3
select_F
<Local:1:I.Data.1>
sensor_S
<Local:2:I.Data.2>
sensor_F
<Local:2:I.Data.1>
move_S_F
move_S_F
4
select_F
<Local:1:I.Data.1>
sensor_T
<Local:2:I.Data.3>
sensor_F
<Local:2:I.Data.1>
move_T_F
move_T_F
5
select_S
<Local:1:I.Data.2>
sensor_G
<Local:2:I.Data.0>
sensor_S
<Local:2:I.Data.2>
move_G_S
move_G_S
6
select_S
<Local:1:I.Data.2>
sensor_F
<Local:2:I.Data.1>
sensor_S
<Local:2:I.Data.2>
move_F_S
move_F_S
7
select_S
<Local:1:I.Data.2>
sensor_T
<Local:2:I.Data.3>
sensor_S
<Local:2:I.Data.2>
move_T_S
move_T_S
8
move_F_S
move_G_F
move_G_S
any_T
down_motor
<Local:1:O.Data.0>
up_motor
<Local:1:O.Data.1>


9
move_S_F
move_T_F
move_T_S
any_G
up_motor
<Local:1:O.Data.1>
down_motor
<Local:1:O.Data.0>
(End)

Experiment - 9

COFFEE VENDING MACHINE

Objective:

To implement the coffee vending machine using ladder diagram in RSLogix5000 and GX Works.

Problem Statement:

The PLC task is to control the coffee vending machine. This includes the filling of sugared milk, then adding decoction and heating the mixture in a common container. This has to flow to an empty cup with required quantity. Proximity sensor is used to sense the presence of the cup. The number of cups filled can also be counted.

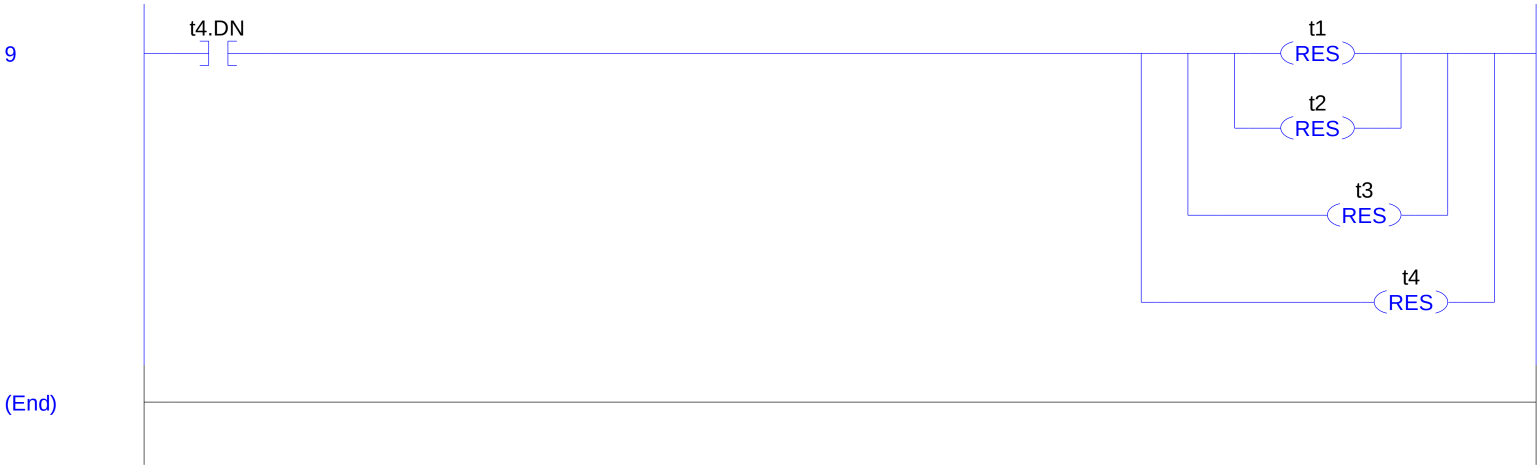
Procedure:

- To develop the coffee vending machine using normally open, normally closed, output coil, timer and counter instructions in the ladder diagram.
- Include the rung comments and rung wise explanation.

Follow up:

Inference: The PLC ladder logic program is created to automate the operations of the coffee vending machine.


START-STOP LOGIC
0
Start
<Local:2:I.Data.0>
Stop
<Local:2:I.Data.1>
Control_Relay
Control_Relay
SENSE THE PRESENCE OF CUP
1
Control_Relay
Sensor
<Local:2:I.Data.3>
TON
Timer On Delay
Timer t1
Preset 5000
Accum 0
EN
DN
FLOW OF MILK
2
t1.TT
Milk
<Local:2:O.Data.0>
3
t1.DN
TON
Timer On Delay
Timer t2
Preset 5000
Accum 0
EN
DN
FLOW OF DECOCTION
4
t2.TT
Decoction
<Local:2:O.Data.1>
5
t2.DN
TON
Timer On Delay
Timer t3
Preset 5000
Accum 0
EN
DN
HEATING TIME DURATION
6
t3.TT
Heat
<Local:2:O.Data.2>
7
t3.DN
TON
Timer On Delay
Timer t4
Preset 5000
Accum 0
EN
DN
FLOW OF COFFEE
8
t4.TT
Coffee_Ready
<Local:2:O.Data.3>


9
t4.DN
t1
RES
t2
RES
t3
RES
t4
RES
(End)

Experiment - 10

TRAFFIC CONTROLLER

Objective:

To implement Traffic Controller using ladder diagram in RSLogix5000 and GX Works.

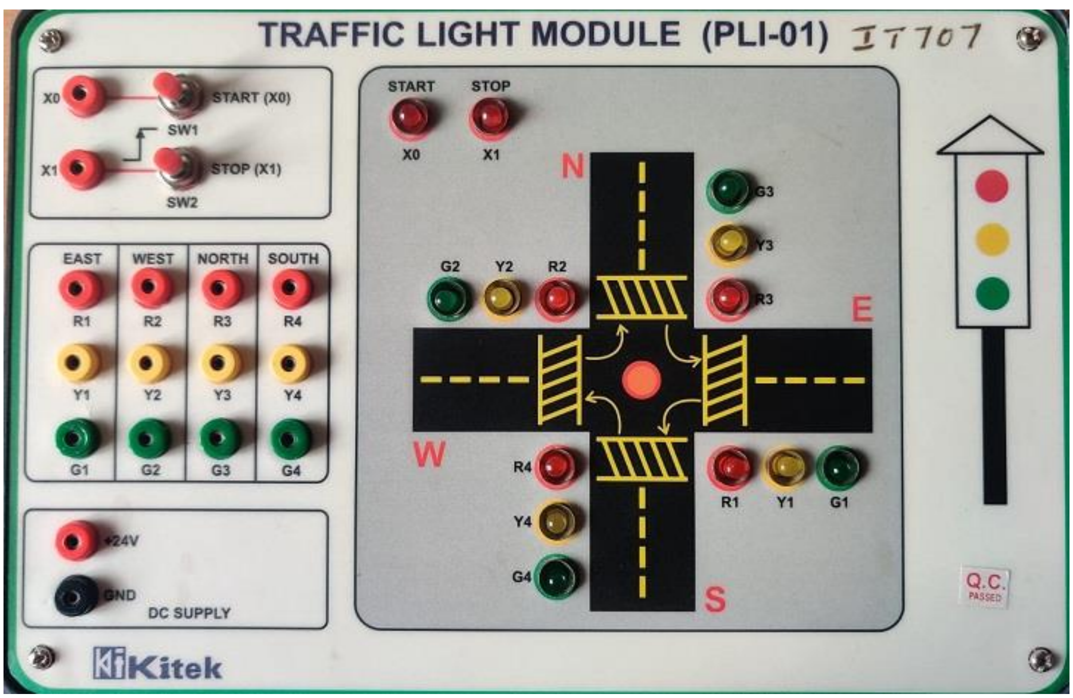
Problem Statement:

The PLC task is to control the traffic. This includes controlling the traffic using Traffic Controller Module.

Procedure:

- To develop traffic control instructions the ladder diagram.
- Include the rung comments and rung wise explanation.

Process Diagram: (Can perform 3way / 4way)



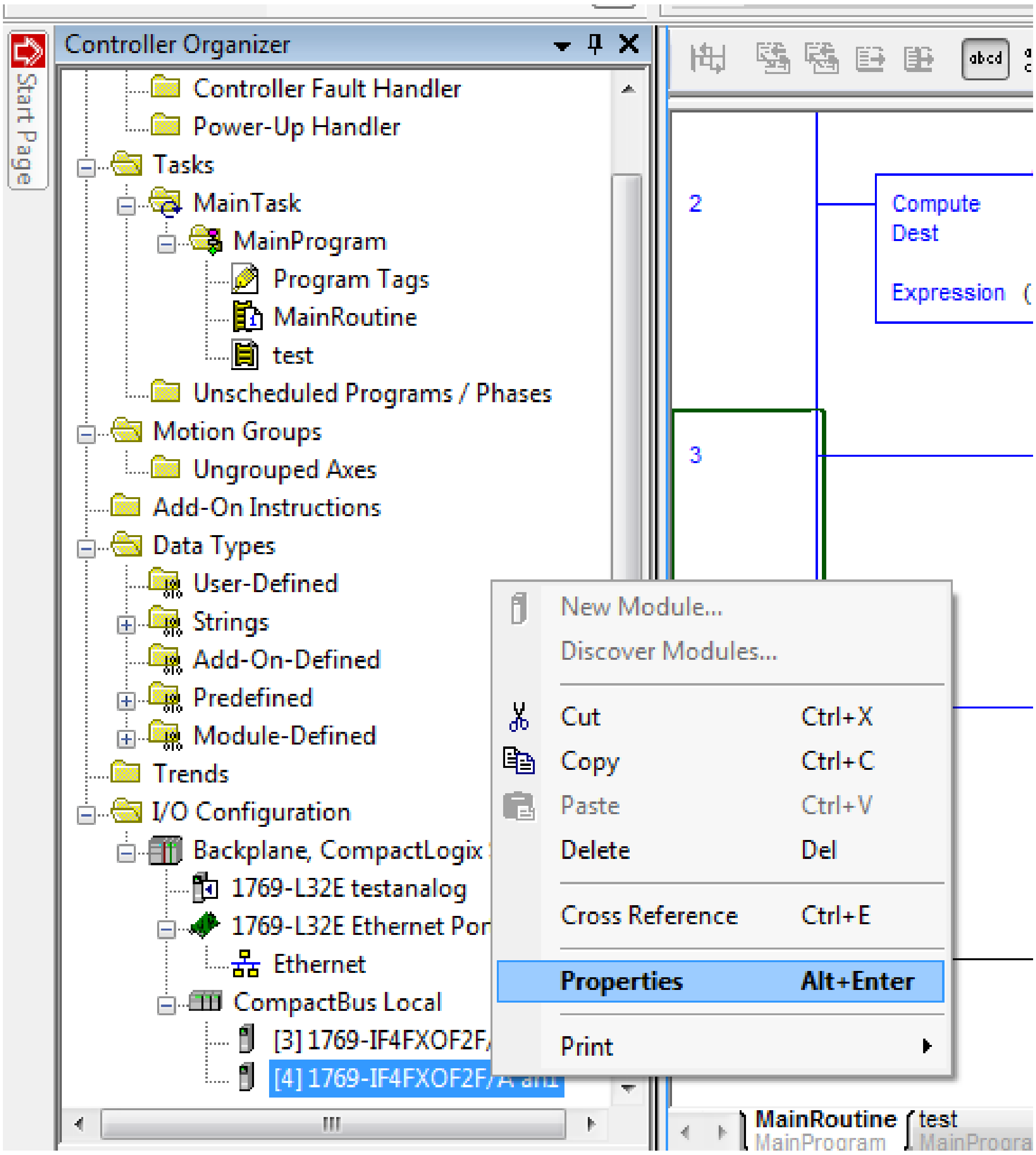
Inference: Implemented the required traffic light controller using PLC ladder logic. Also, demonstrated the traffic control using the traffic controller module.

Configuration of Analog module:

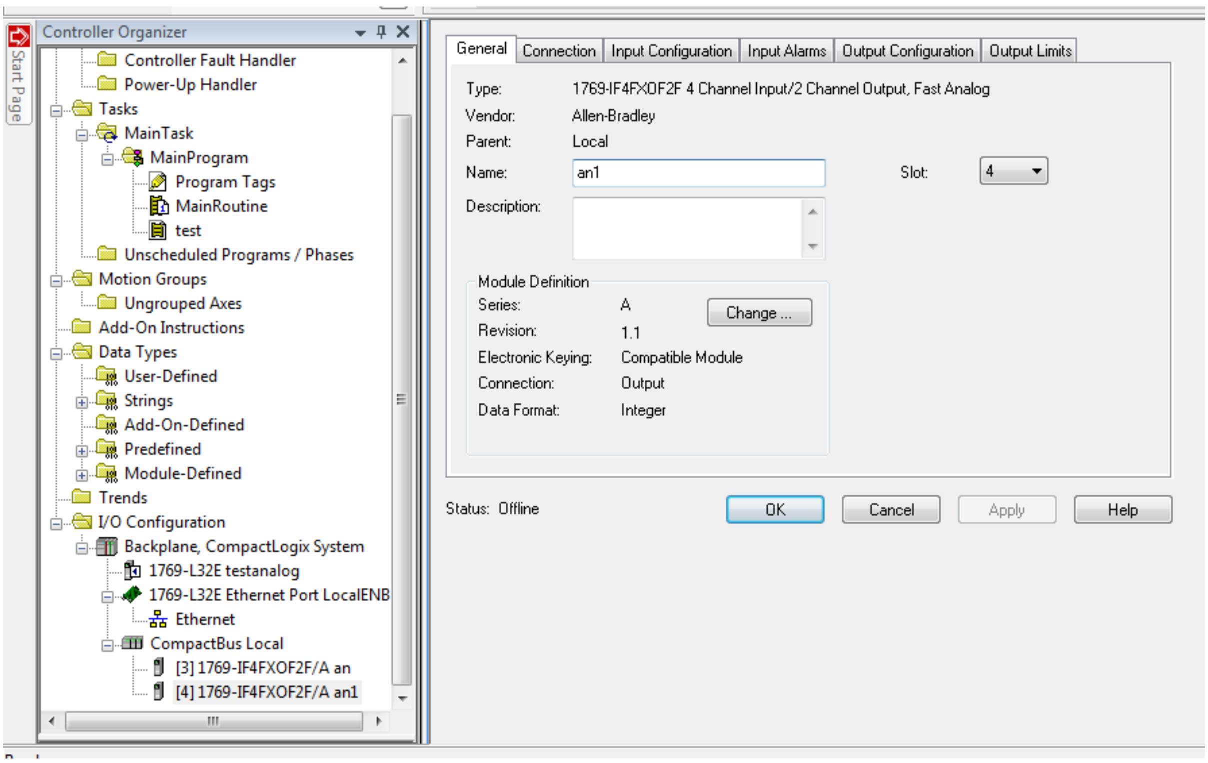
Module Name: 1769IF4FXOF2F

Steps to configure:

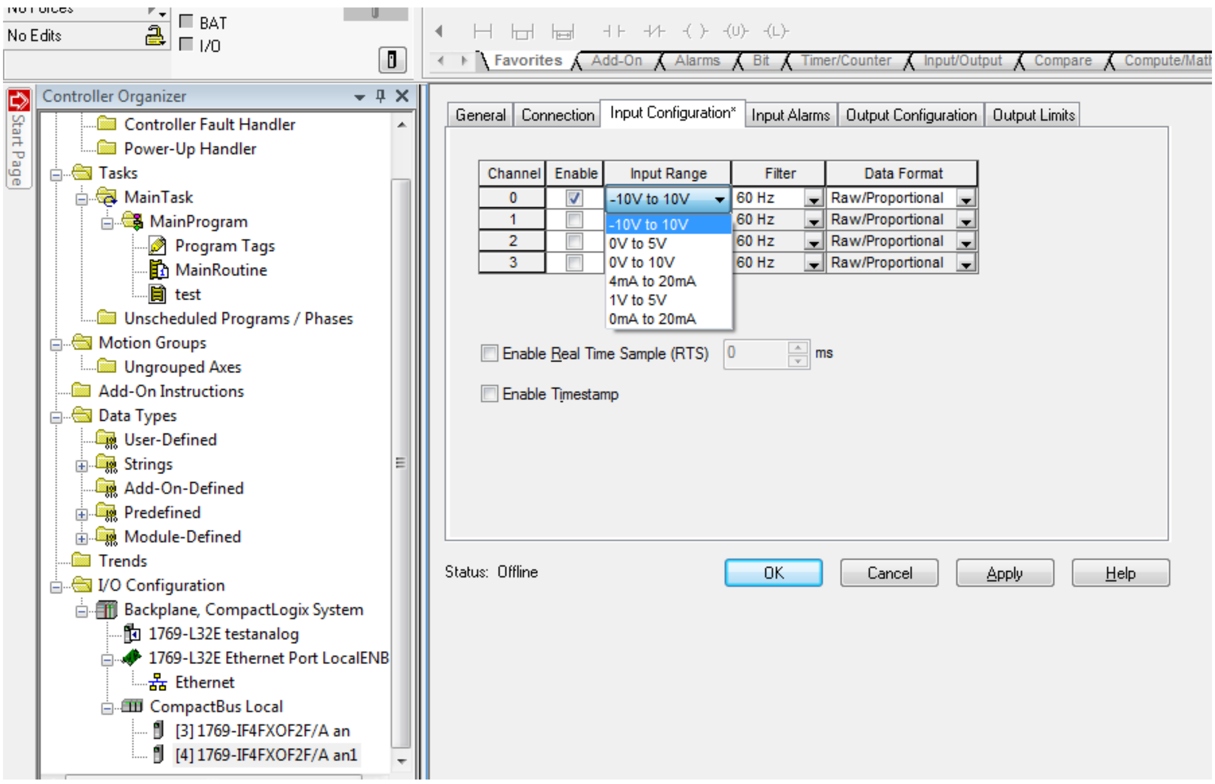
1. Right click on Compact Bus Local in controller organizer window
2. Select catalogue number 1769IF4FXOF2F and select create
3. Enter name for module and select slot number 3 or 4
4. Select the properties option as shown below.

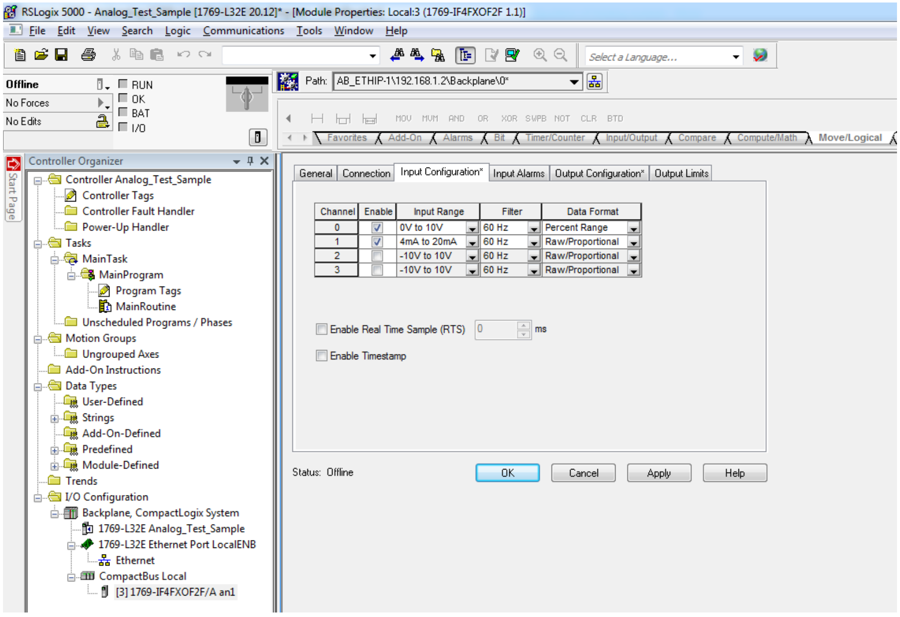


5. The following window will be opened and select input configuration and output configuration

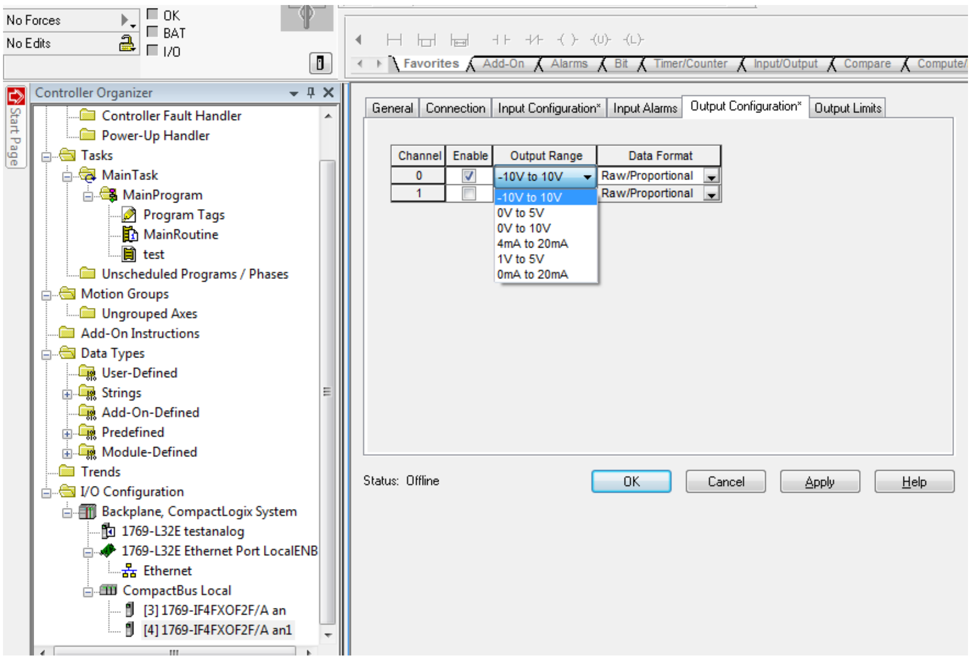


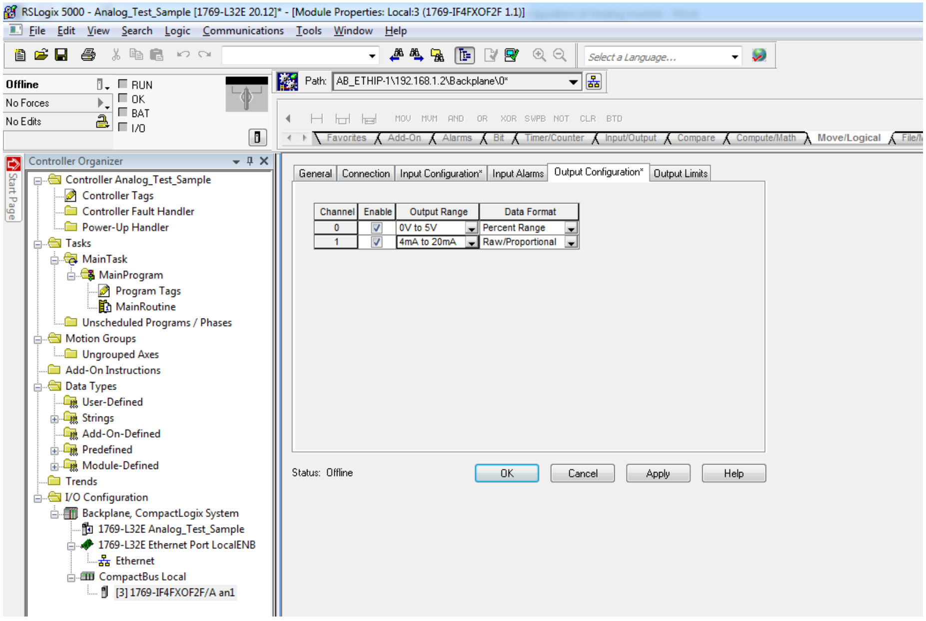
6. Selection of channels should be performed as given





7. Here Input range: 0v to 5v, 0v to 10v, 4mA to 20 mA, 1v to 5 v and 0mA to 20 mA are listed. We should select any one as per the input device/input values to be provided.
   If Input is Voltage, voltage ranges should be selected for current, current range of values should be selected. Data format can be Raw/proportional or Percentage.
8. Similarly output configuration has to be made.





9. Output limits can be set from 0 to 32764 or 10000

Illustration for Reading Analog input voltage and Displaying Analog output voltage (using Ladder Diagram):

Objective: Read and write voltage using Analog module

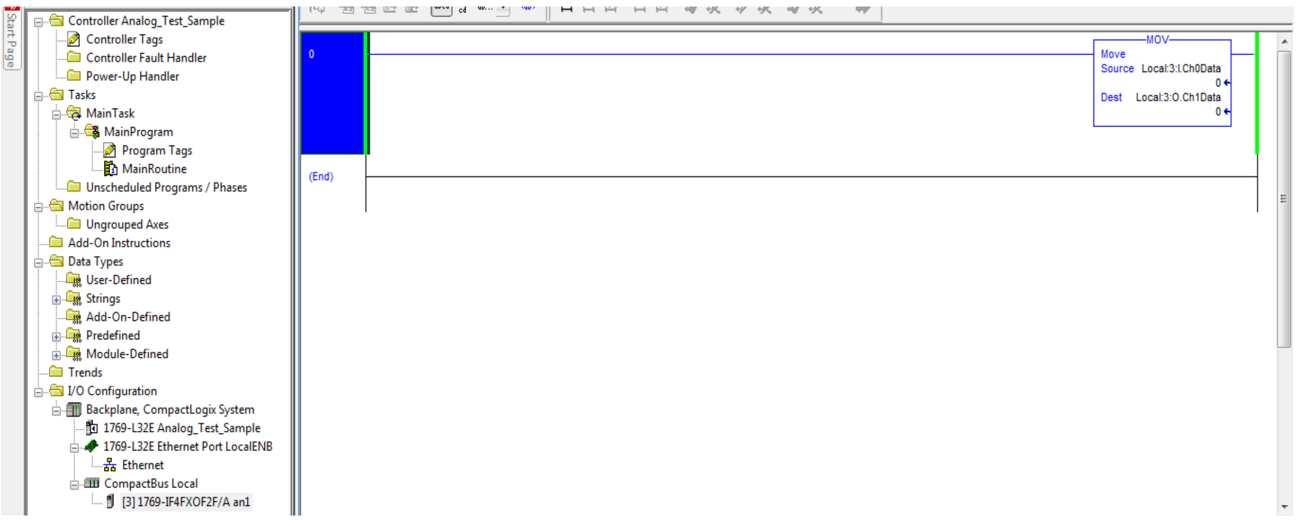
Steps:

Create a ladder, Use MOV instruction, In Mov instruction two parameters such as Source and Dest is available.

Select Input channel Local:3:I: Ch0Data for source. This reads the analog voltage data.

Select Output channel Local:3:O: Ch0Data for Dest. This writes the analog voltage data.

Ladder Diagram:



Connection in Hardware module:

Connect power supply with (1-10 V range) Positive terminal to VinO+ (input) and Negative(Gnd) to Analog common in Module3 (slot3).

For each voltage input given from power supply, the Mov instruction will read and display the scaled values. The approximate scaled values are listed in the table below: (this is listed for 0-10v with output limits 0-10000 scaling value. This might change for different scaling limits.)

Voltage (v)	Scaled value
1	500
2	1000
3	1500
4	2000
5	2500
6	3000
7	3500
8	4000
9	4500
10	5000

Inference: Voltage is acquired and displayed using analog module.

Follow up: Connect any sensor which gives voltage output. Read the values and display.

Illustration of Reading Analog input Current and Displaying Analog output current (using Objective: Read and write Current using Analog module

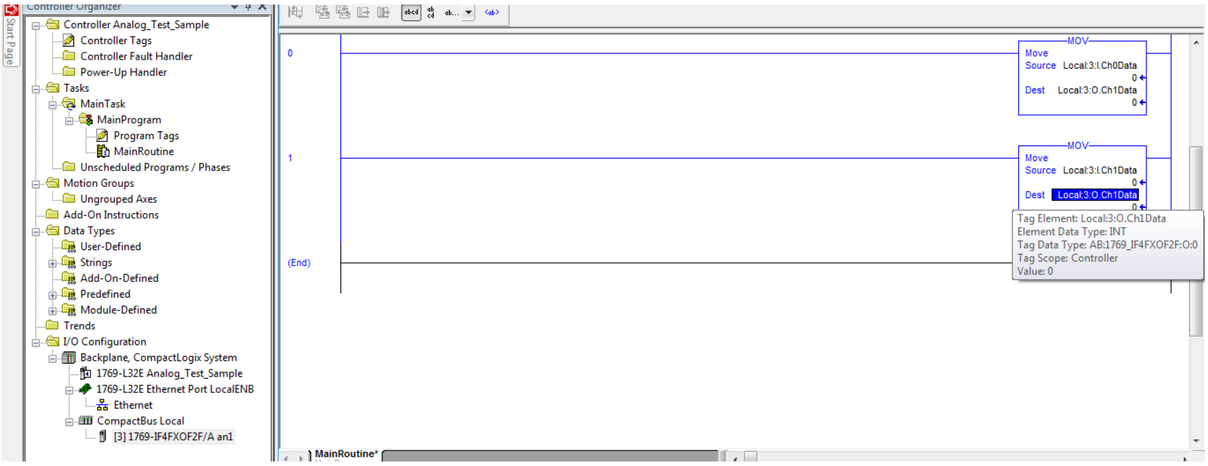
Steps:

Create a ladder, Use MOV instruction, In Mov instruction two parameters such as Source and Dest is available.

Select Input channel Local:3:I: Ch1Data for source. This reads the Analog Current data.

Select Output channel Local:3:O: Ch1Data for Dest. This writes the Analog Current data.

Ladder Diagram:



Connection in Hardware module:

Connect current source with (4-20mA range) Positive terminal to Iin1+ (input) and Negative(Gnd) to V/I 1-.

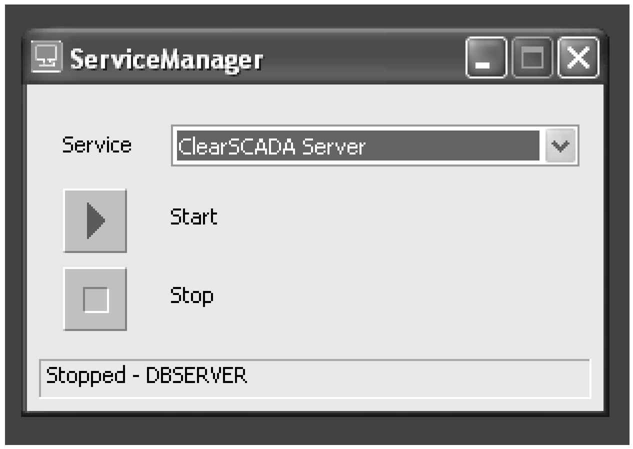
For each current input given from current cource, the Mov instruction will read and display the scaled values. The approximate scaled values are listed in the table below: (this is listed for 4-20mA with output limits 0-10000 scaling value. This might change for different scaling limits.) Here output will be viewed using multimeter.

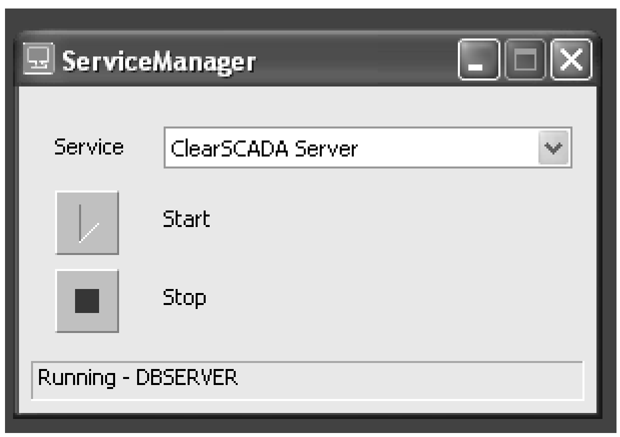
Current (mA)	Scaled value
4	100
7	2000
8	3000
10	4000
14	6000
16	8000
18	9000
20	10000

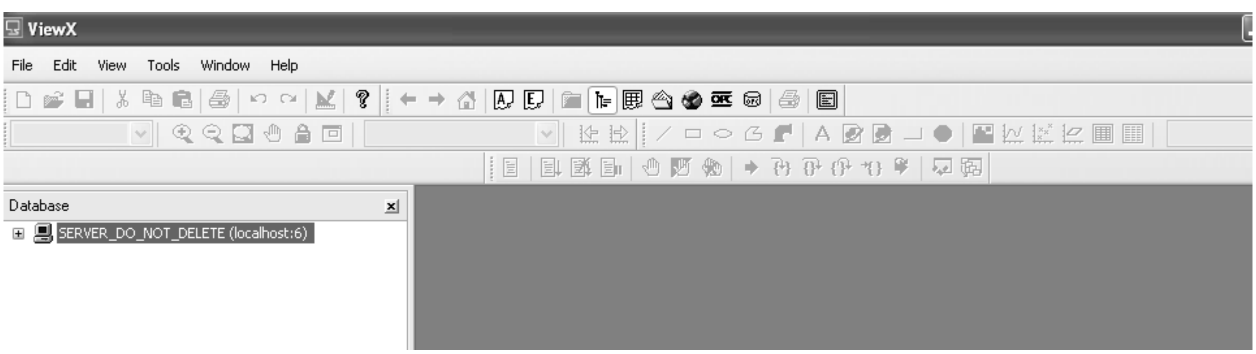
Inference: Current is acquired and displayed using analog module

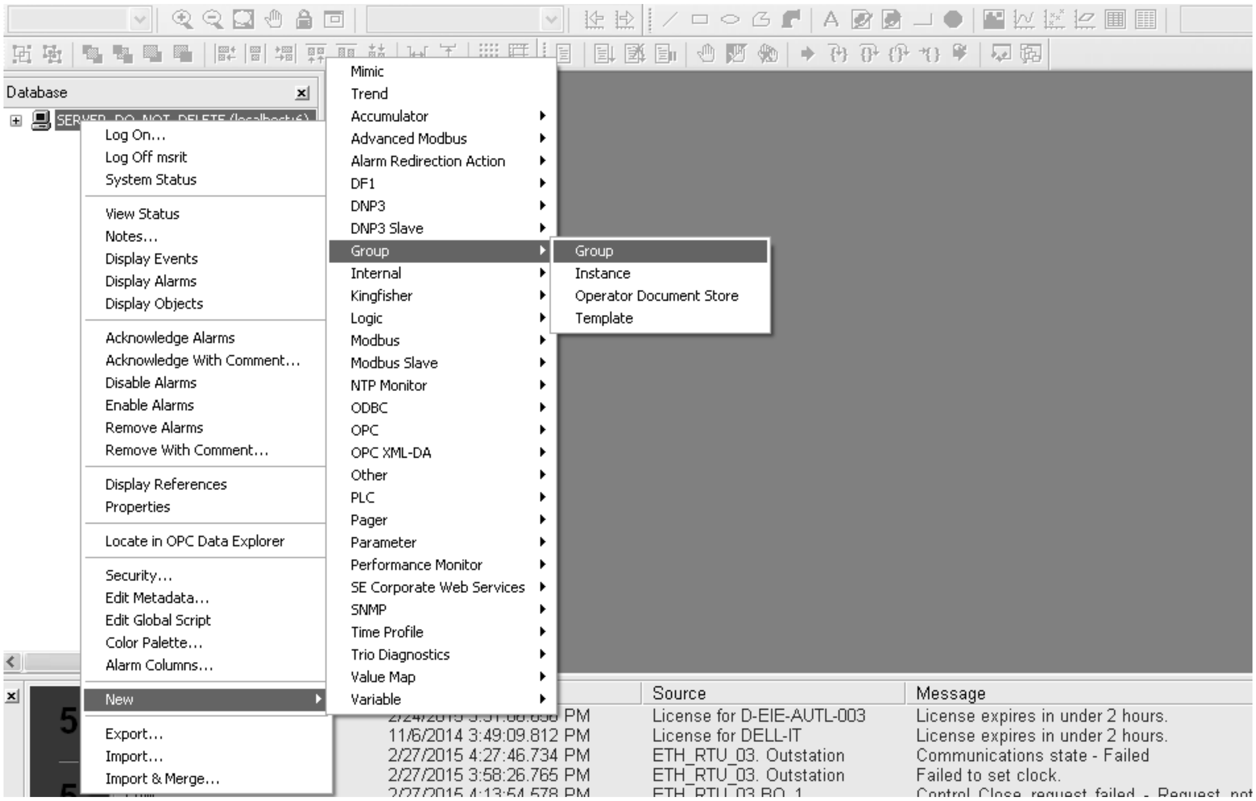
Follow up: Connect any sensor which aids in reading and displaying current values.

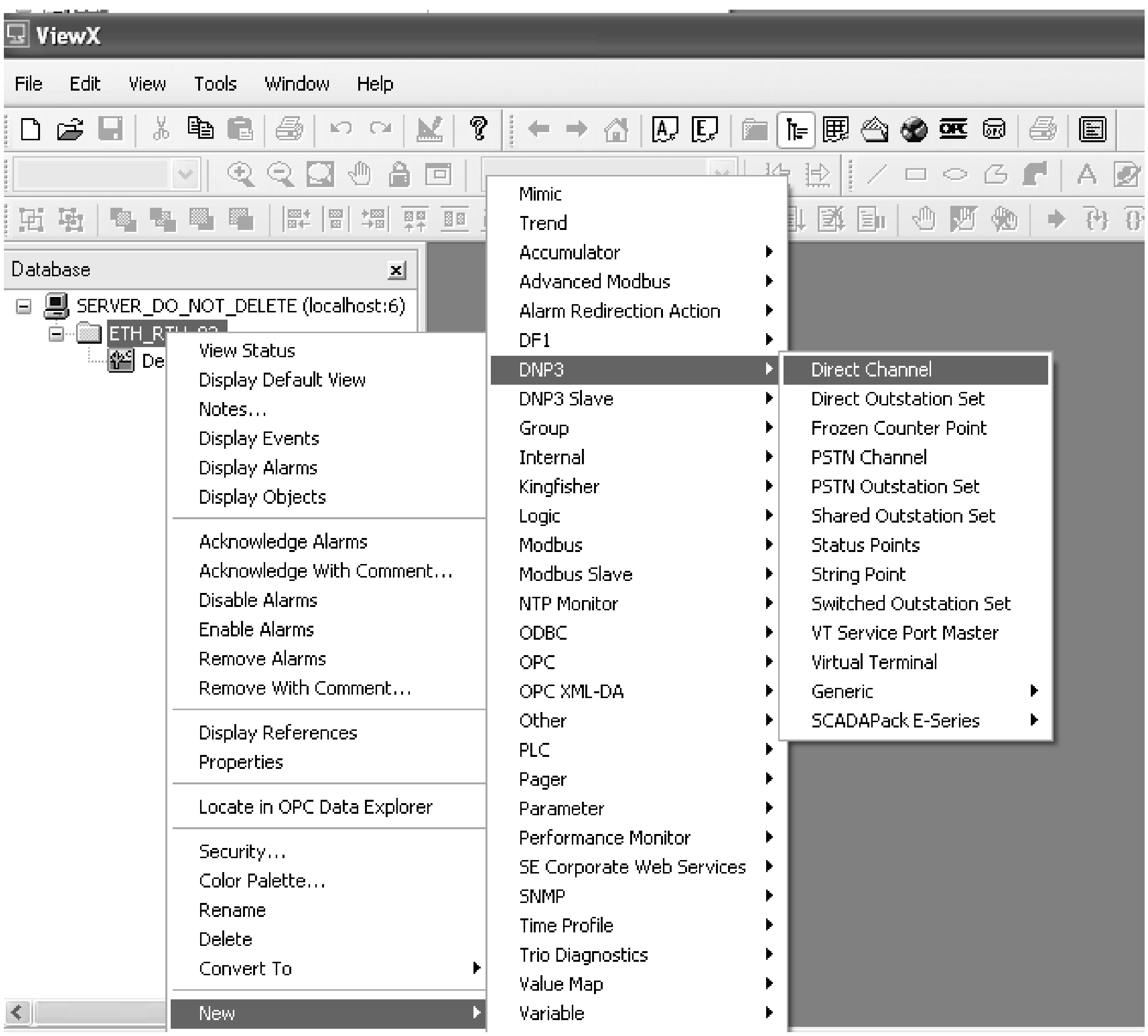
Procedure for Configuring the SCADA System and Creation of the Ladder Logic in ClearSCADA

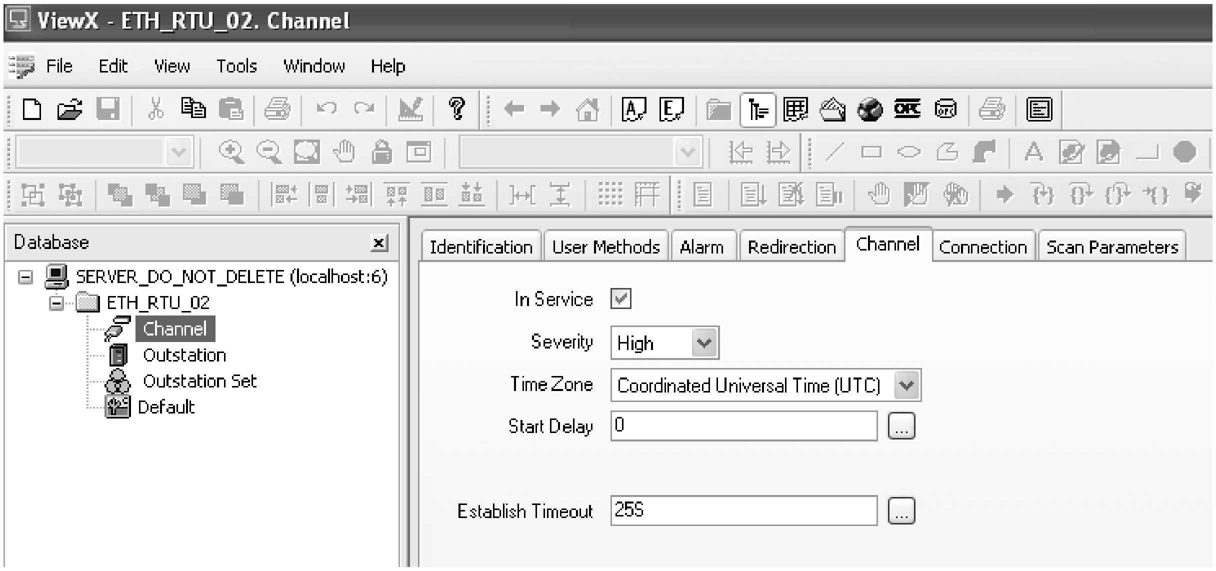


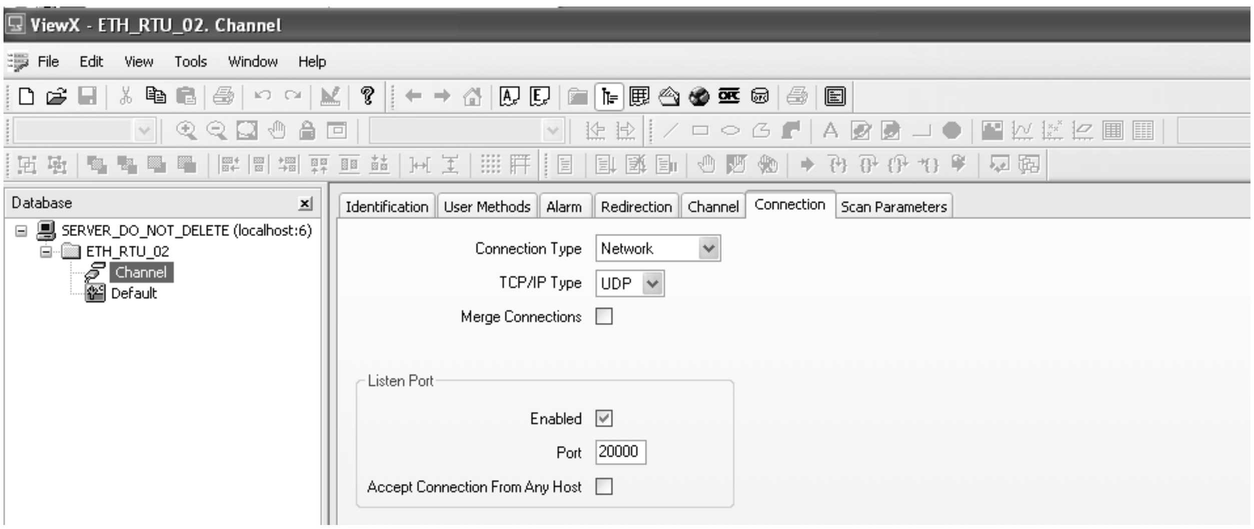


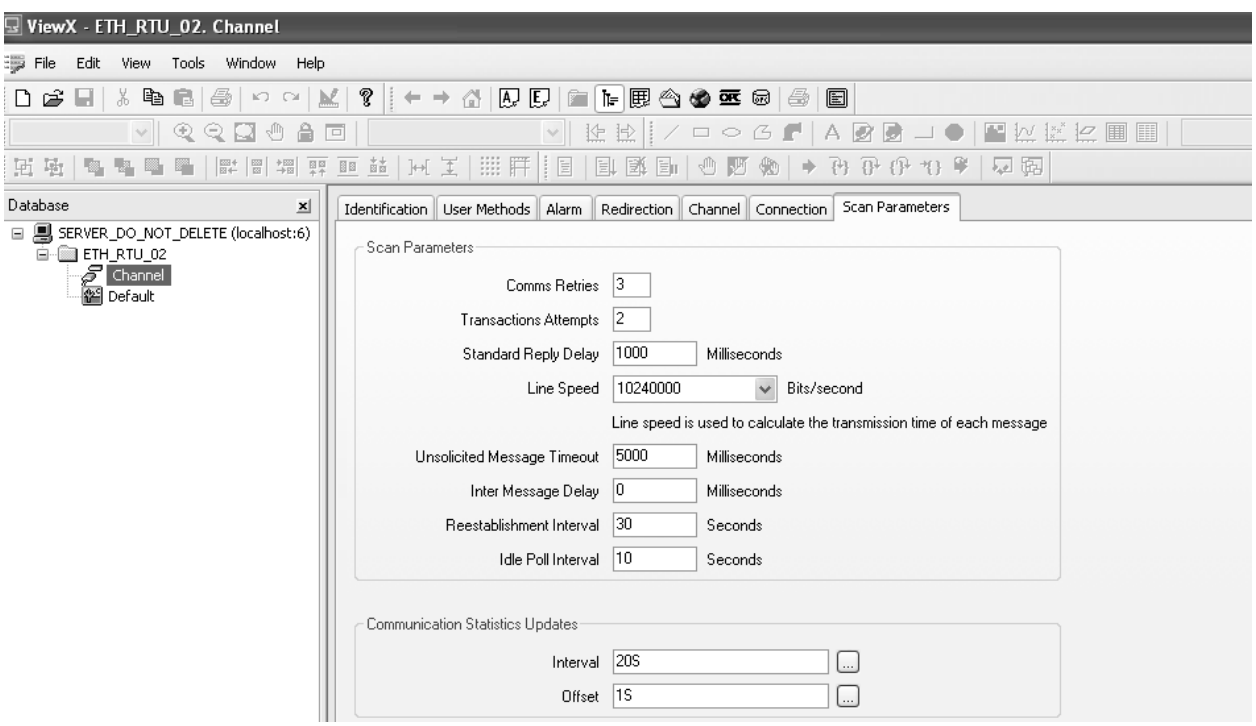


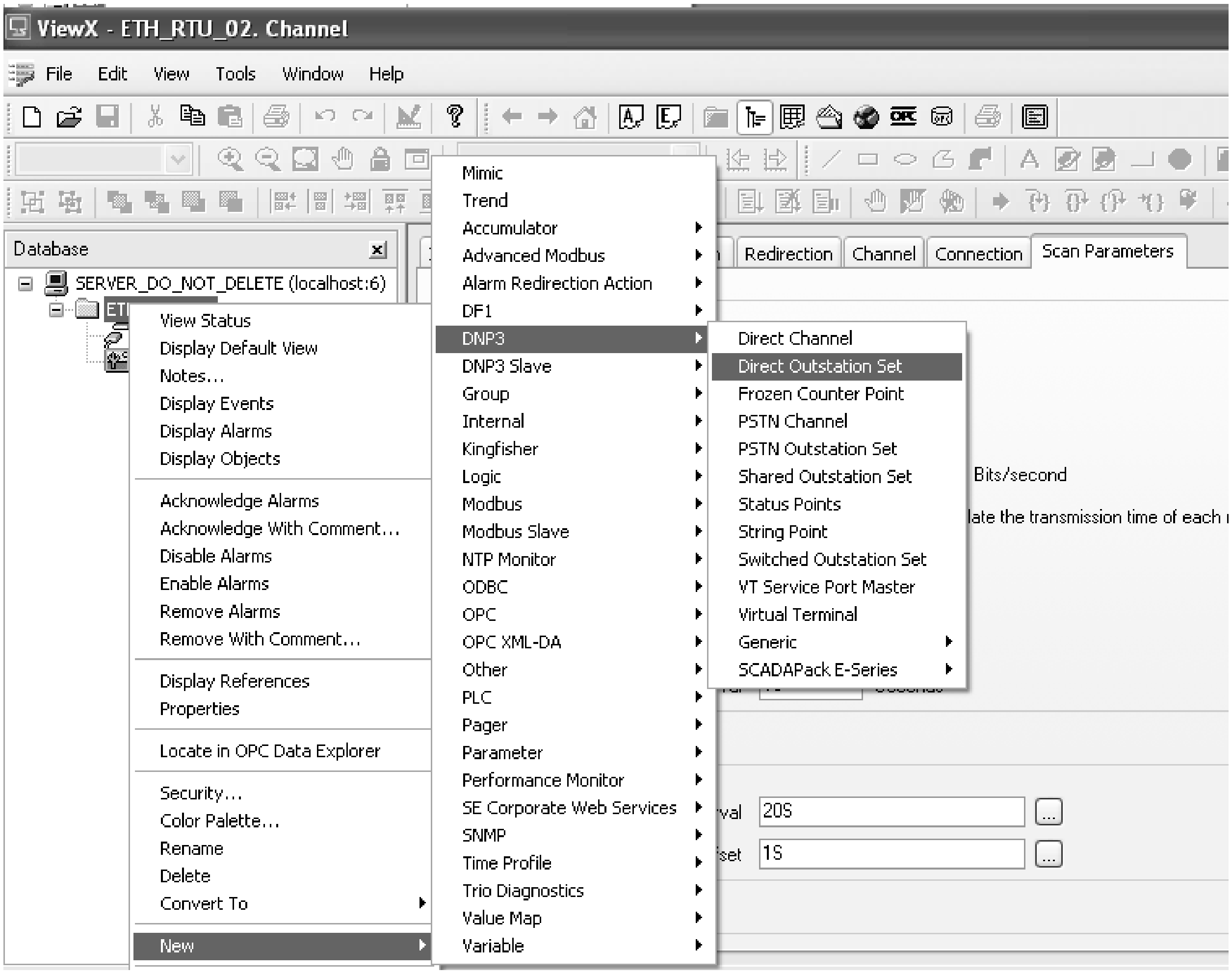

Database
Log On...
Log Off msrit
System Status
View Status
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Edit Metadata...
Edit Global Script
Color Palette...
Alarm Columns...
New
Export...
Import...
Import & Merge...
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Group
Instance
Operator Document Store
Template
Source
Message
11/6/2014 3:49:09.812 PM
License for D-EIE-AUTL-003
License for DELL-IT
License expires in under 2 hours.
License expires in under 2 hours.
2/27/2015 4:27:46.734 PM
ETH_RTU_03. Outstation
Communications state - Failed
2/27/2015 3:58:26.765 PM
ETH_RTU_03. Outstation
Failed to set clock.

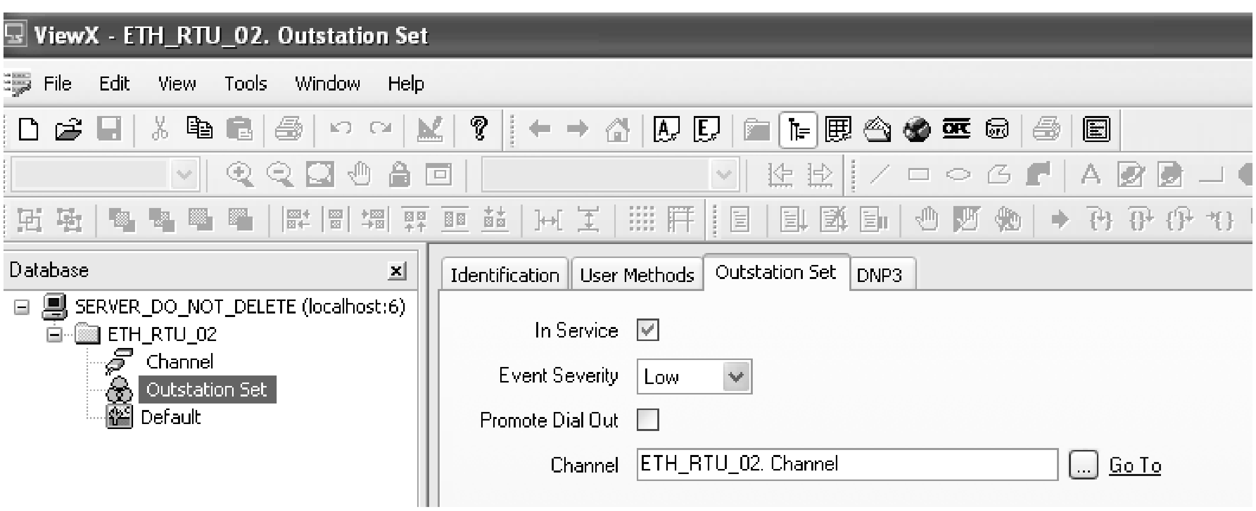

ViewX
File
Edit
View
Tools
Window
Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Direct Channel
Direct Outstation Set
Frozen Counter Point
PSTN Channel
PSTN Outstation Set
Shared Outstation Set
Status Points
String Point
Switched Outstation Set
VT Service Port Master
Virtual Terminal
Generic
SCADAPack E-Series

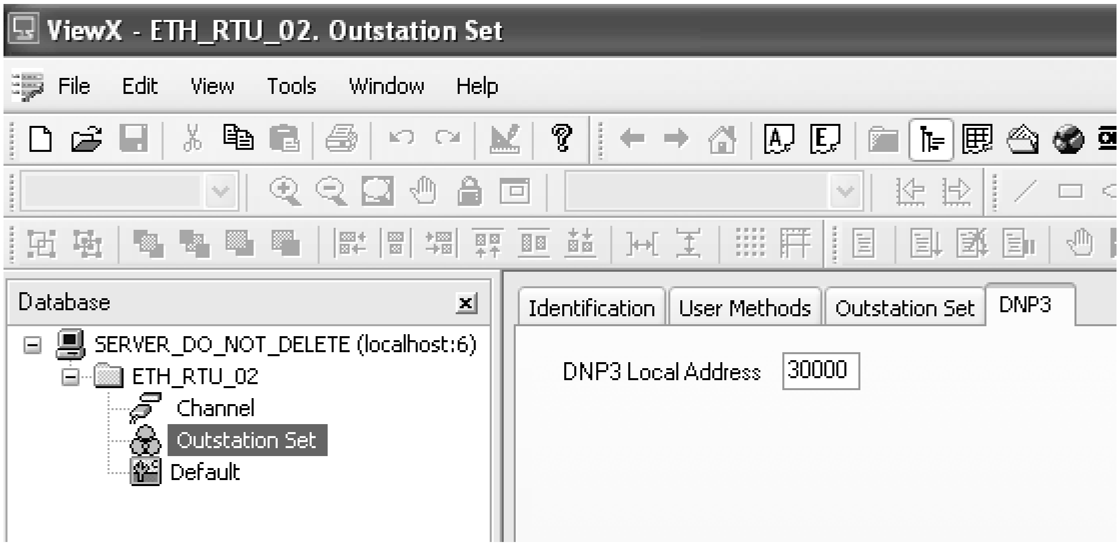

ViewX - ETH_RTU_02. Channel
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
Default
Identification User Methods Alarm Redirection Channel Connection Scan Parameters
In Service
Severity High
Time Zone Coordinated Universal Time (UTC)
Start Delay 0
Establish Timeout 25S

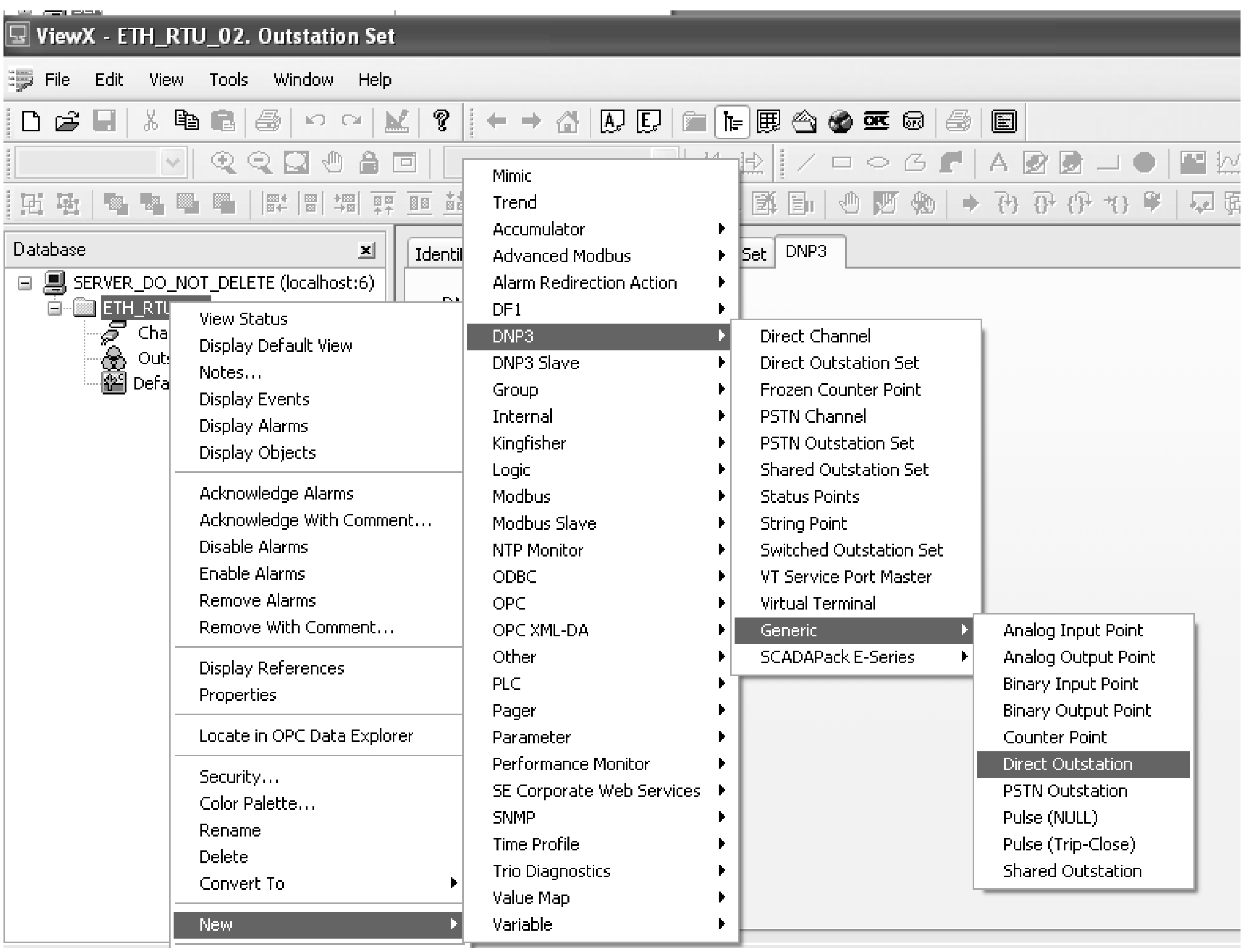

ViewX - ETH_RTU_02. Channel
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Default
Identification User Methods Alarm Redirection Channel Connection Scan Parameters
Connection Type Network
TCP/IP Type UDP
Merge Connections
Listen Port
Enabled
Port 20000
Accept Connection From Any Host

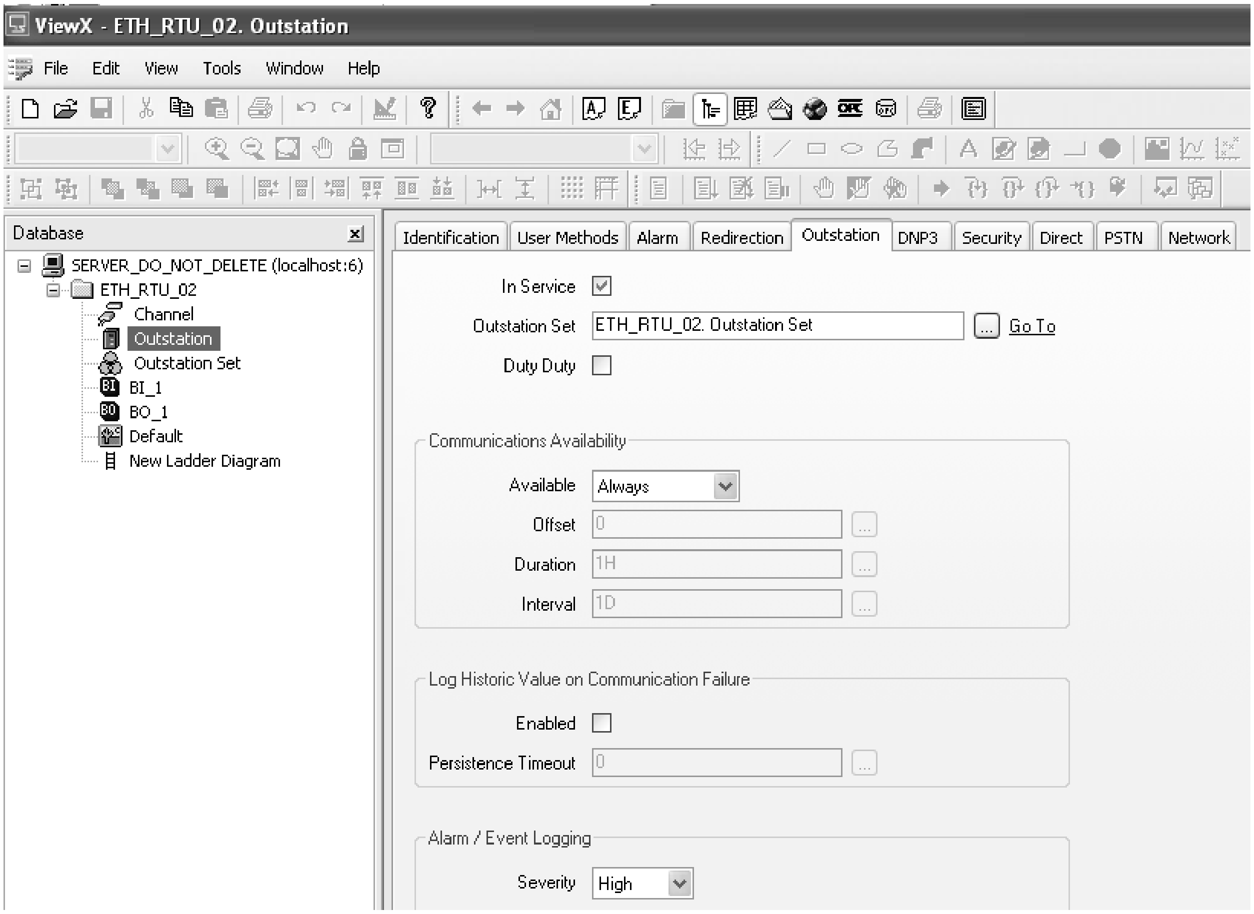

ViewX - ETH_RTU_02. Channel
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Default
Identification User Methods Alarm Redirection Channel Connection Scan Parameters
Scan Parameters
Comms Retries 3
Transactions Attempts 2
Standard Reply Delay 1000 Milliseconds
Line Speed 10240000 Bits/second
Line speed is used to calculate the transmission time of each message
Unsolicited Message Timeout 5000 Milliseconds
Inter Message Delay 0 Milliseconds
Reestablishment Interval 30 Seconds
Idle Poll Interval 10 Seconds
Communication Statistics Updates
Interval 20S
Offset 1S

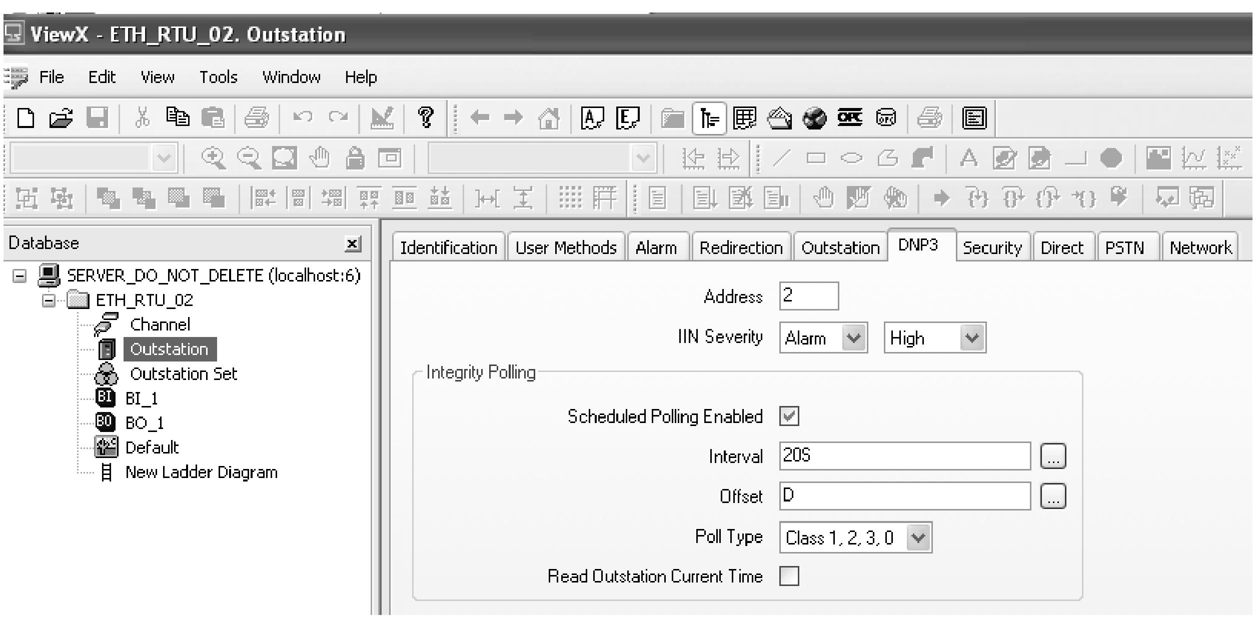

ViewX - ETH_RTU_02. Channel
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Direct Channel
Direct Outstation Set
Frozen Counter Point
PSTN Channel
PSTN Outstation Set
Shared Outstation Set
Status Points
String Point
Switched Outstation Set
VT Service Port Master
Virtual Terminal
Generic
SCADAPack E-Series
Redirection
Channel
Connection
Scan Parameters
Bits/second
late the transmission time of each
20S
1S

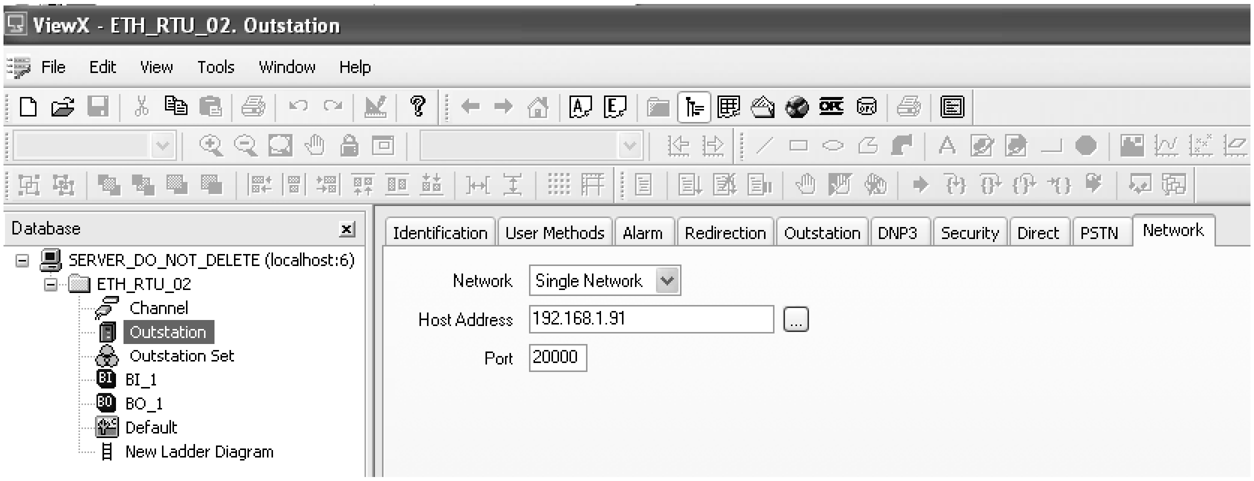

ViewX - ETH_RTU_02. Outstation Set
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation Set
Default
Identification
User Methods
Outstation Set
DNP3
In Service
Event Severity
Low
Promote Dial Out
Channel
ETH_RTU_02. Channel
Go To

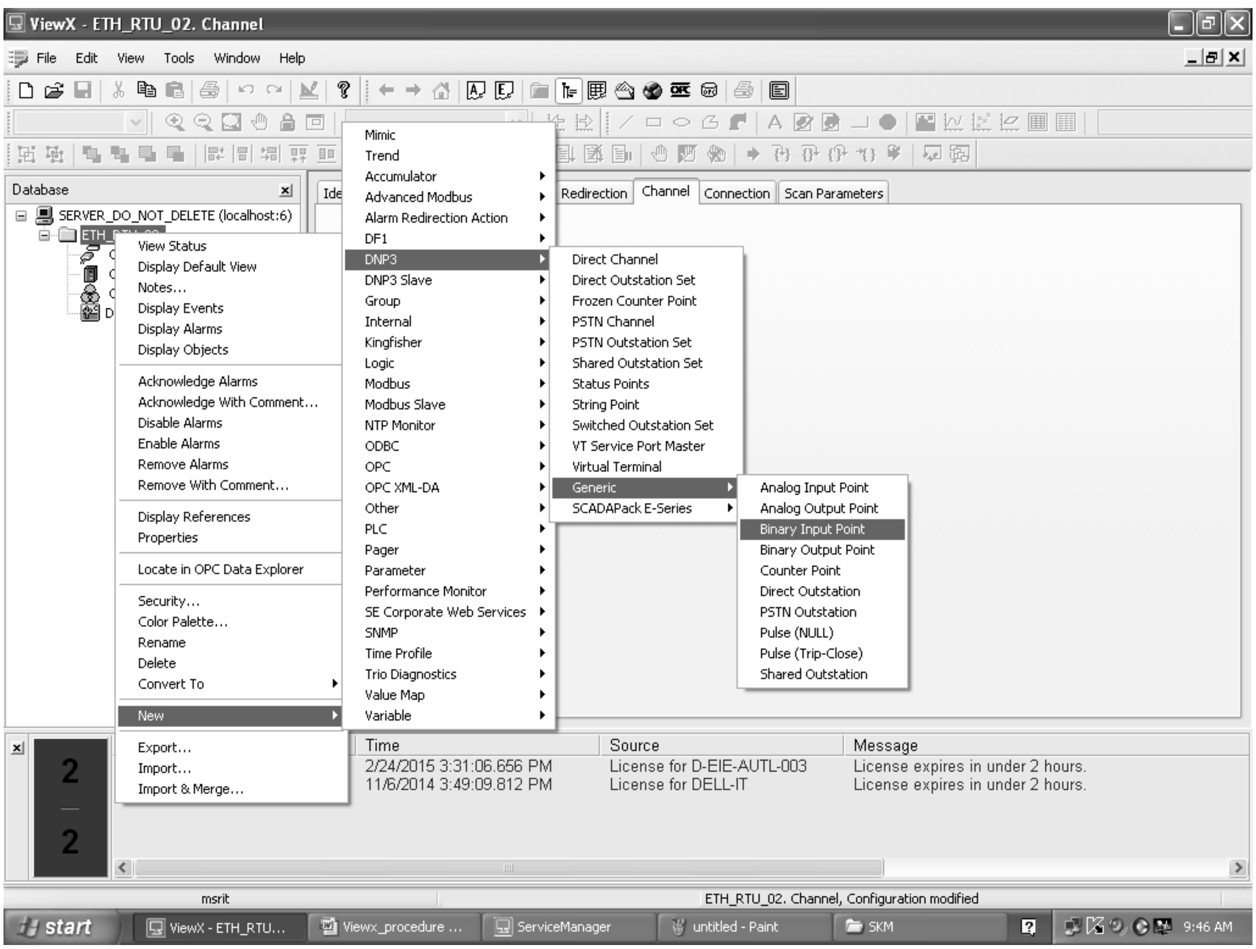

ViewX - ETH_RTU_02. Outstation Set
File
Edit
View
Tools
Window
Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation Set
Default
Identification
User Methods
Outstation Set
DNP3
DNP3 Local Address
30000

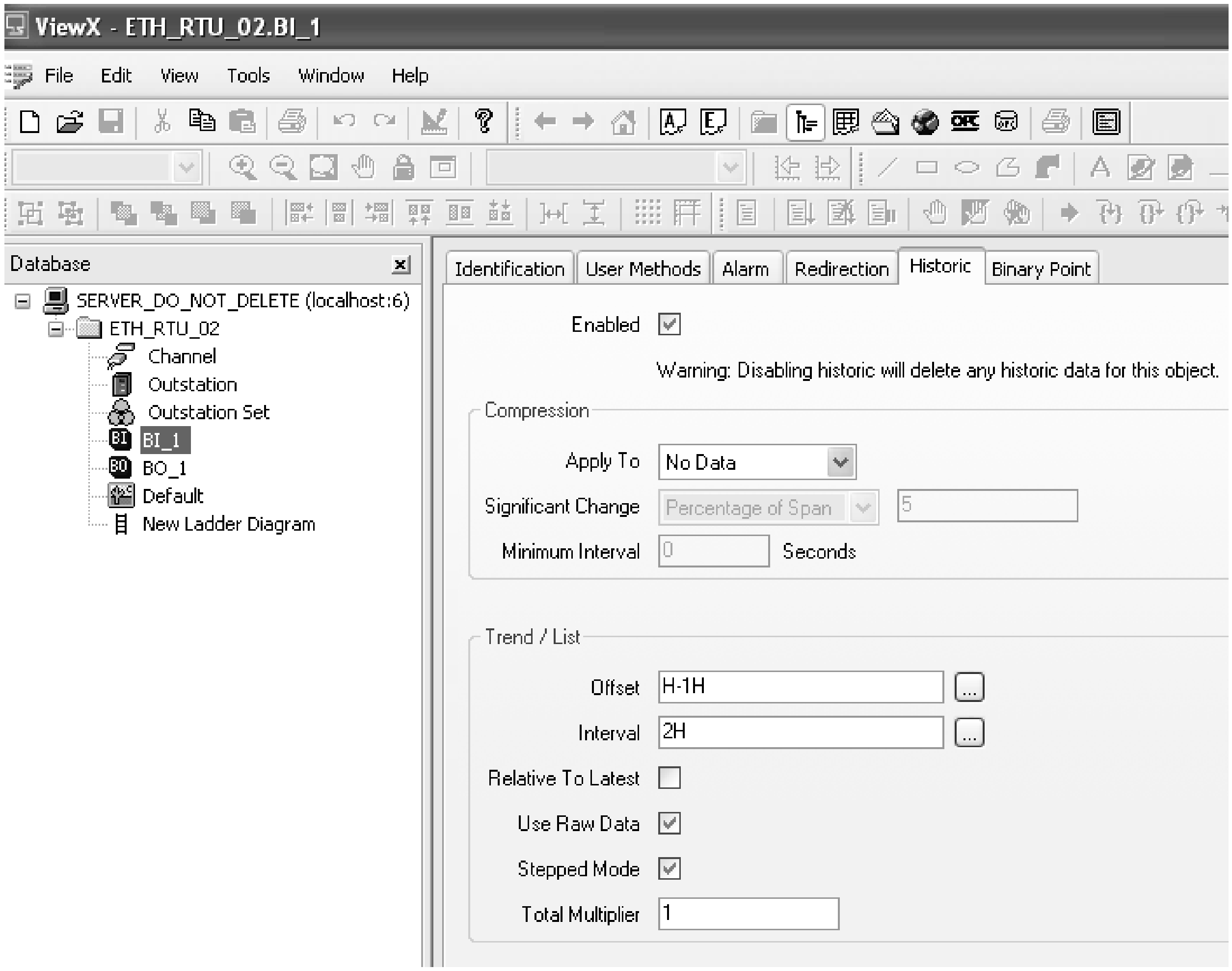
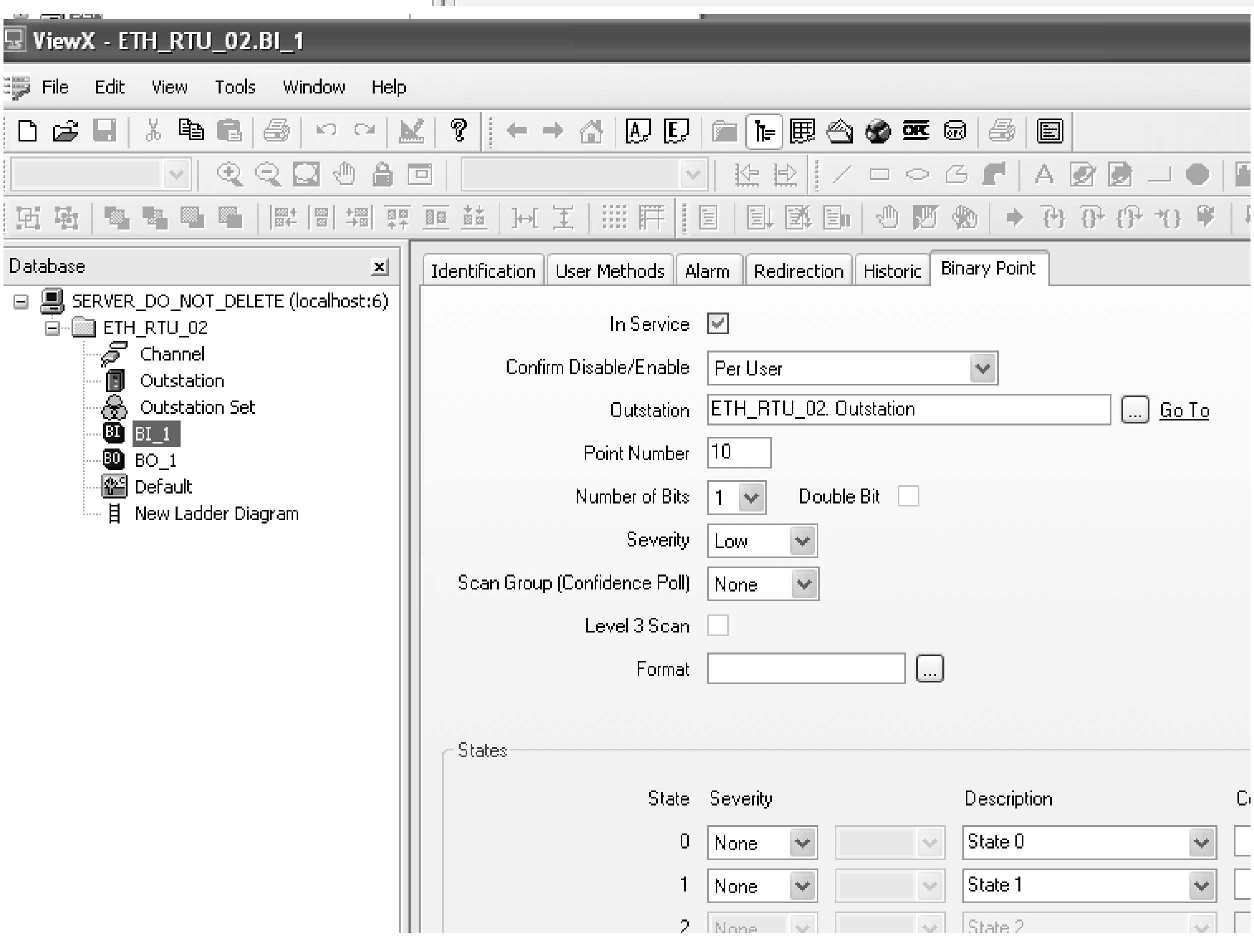

ViewX - ETH_RTU_02. Outstation Set
File
Edit
View
Tools
Window
Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Direct Channel
Direct Outstation Set
Frozen Counter Point
PSTN Channel
PSTN Outstation Set
Shared Outstation Set
Status Points
String Point
Switched Outstation Set
VT Service Port Master
Virtual Terminal
Generic
SCADAPack E-Series
Analog Input Point
Analog Output Point
Binary Input Point
Binary Output Point
Counter Point
Direct Outstation
PSTN Outstation
Pulse (NULL)
Pulse (Trip-Close)
Shared Outstation
DNP3

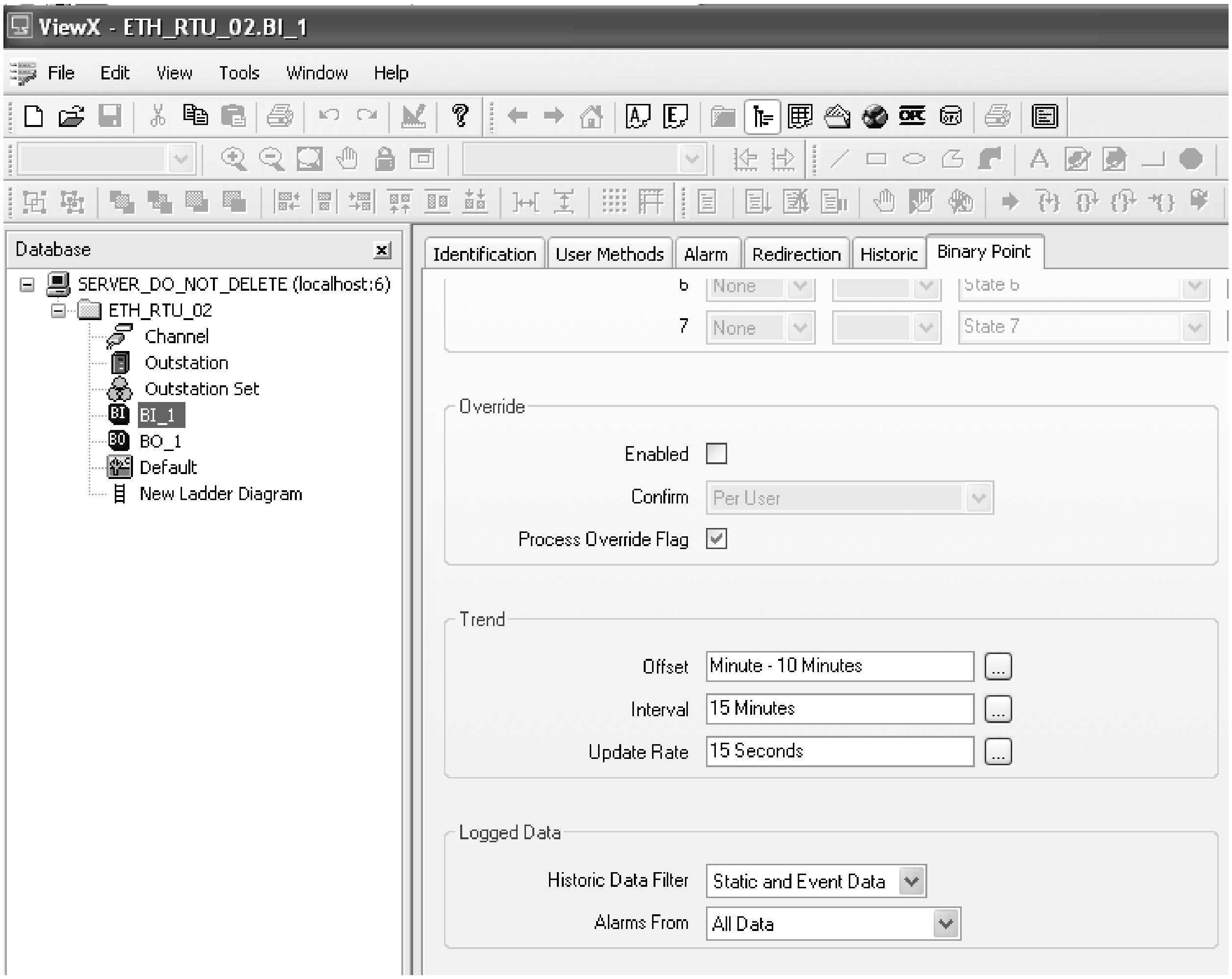

ViewX - ETH_RTU_02. Outstation
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Outstation DNP3 Security Direct PSTN Network
In Service
Outstation Set ETH_RTU_02. Outstation Set
Go To
Duty Duty
Communications Availability
Available Always
Offset 0
Duration 1H
Interval 1D
Log Historic Value on Communication Failure
Enabled
Persistence Timeout 0
Alarm / Event Logging
Severity High

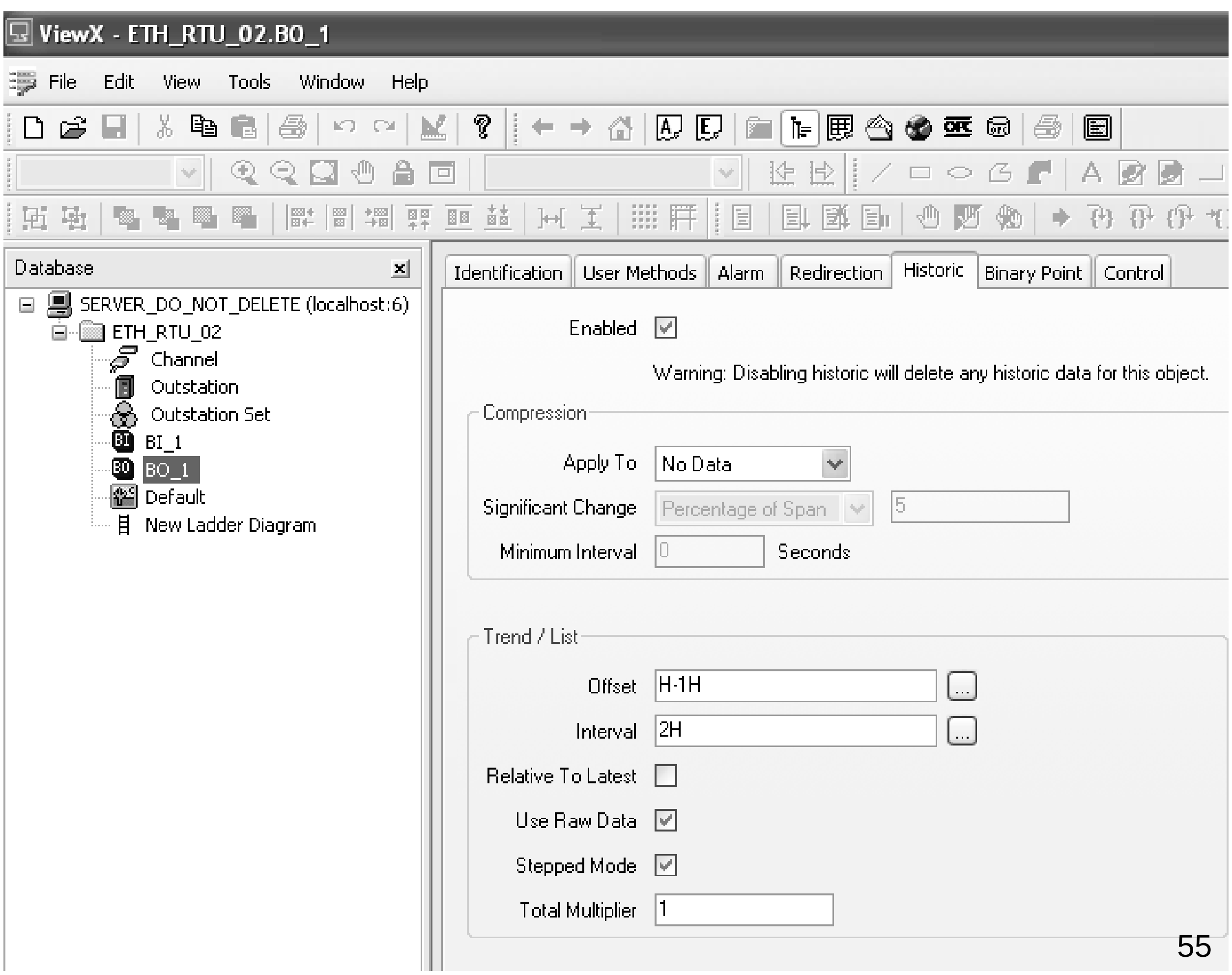

ViewX - ETH_RTU_02. Outstation
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Outstation DNP3 Security Direct PSTN Network
Address 2
IIN Severity Alarm High
Integrity Polling
Scheduled Polling Enabled
Interval 20S
Offset D
Poll Type Class 1, 2, 3, 0
Read Outstation Current Time

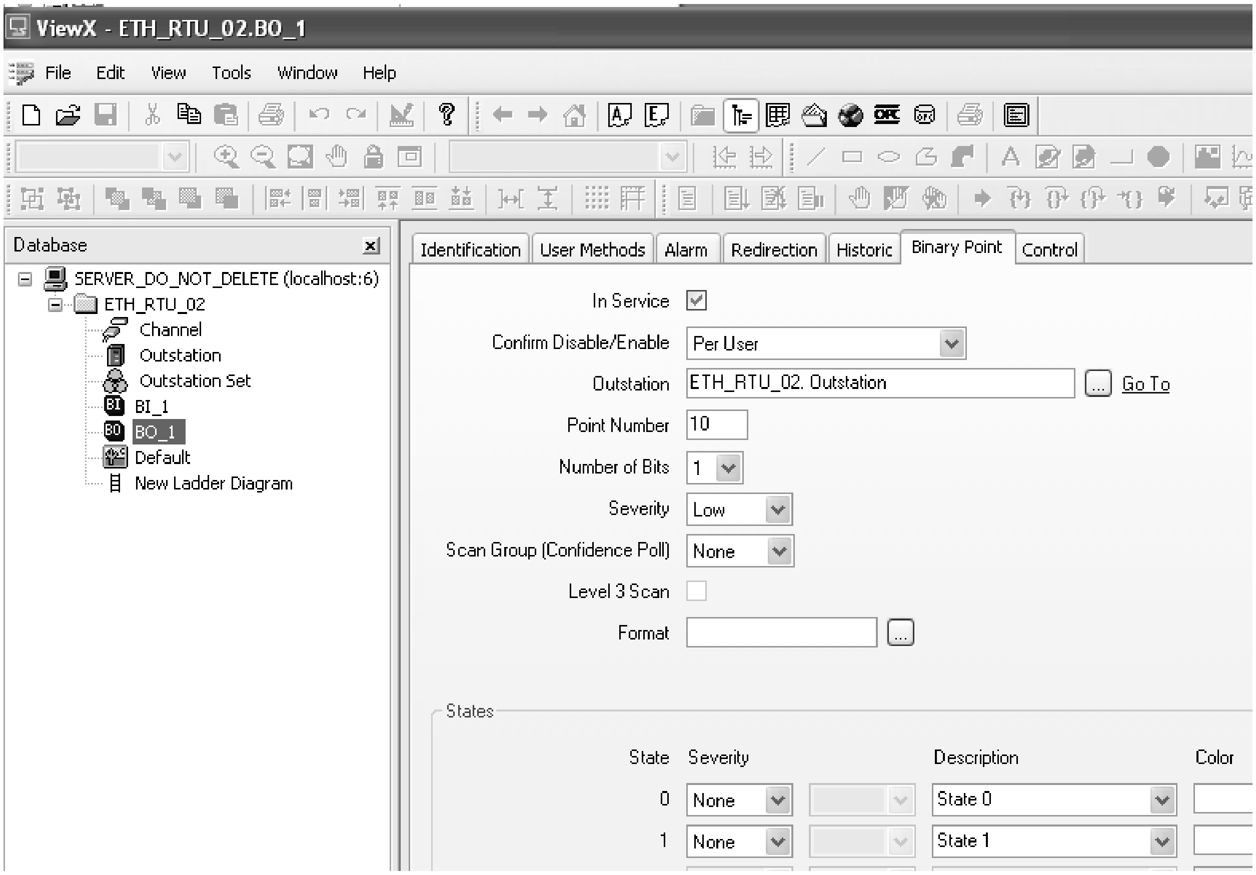

ViewX - ETH_RTU_02. Outstation
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Outstation DNP3 Security Direct PSTN Network
Network Single Network
Host Address 192.168.1.91
Port 20000

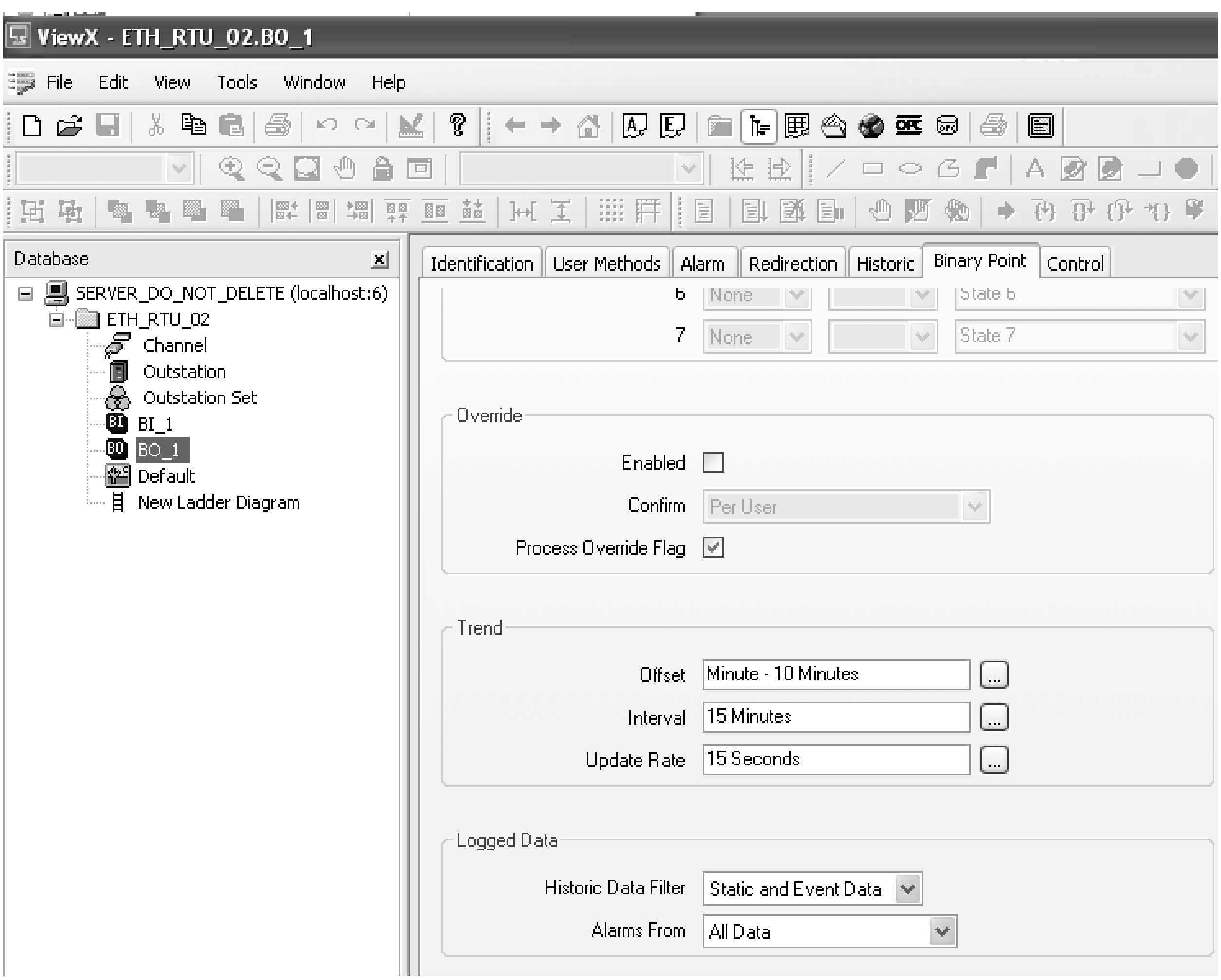

ViewX - ETH_RTU_02. Channel
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Export...
Import...
Import & Merge...
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Direct Channel
Direct Outstation Set
Frozen Counter Point
PSTN Channel
PSTN Outstation Set
Shared Outstation Set
Status Points
String Point
Switched Outstation Set
VT Service Port Master
Virtual Terminal
Generic
SCADAPack E-Series
Analog Input Point
Analog Output Point
Binary Input Point
Binary Output Point
Counter Point
Direct Outstation
PSTN Outstation
Pulse (NULL)
Pulse (Trip-Close)
Shared Outstation
Redirection Channel Connection Scan Parameters
Time Source Message
2/24/2015 3:31:06.656 PM License for D-EIE-AUTL-003 License expires in under 2 hours.
11/6/2014 3:49:09.812 PM License for DELL-IT License expires in under 2 hours.
msrit
ETH_RTU_02. Channel, Configuration modified
start
ViewX - ETH_RTU...
Viewx_procedure ...
ServiceManager
untitled - Paint
SKM
9:46 AM

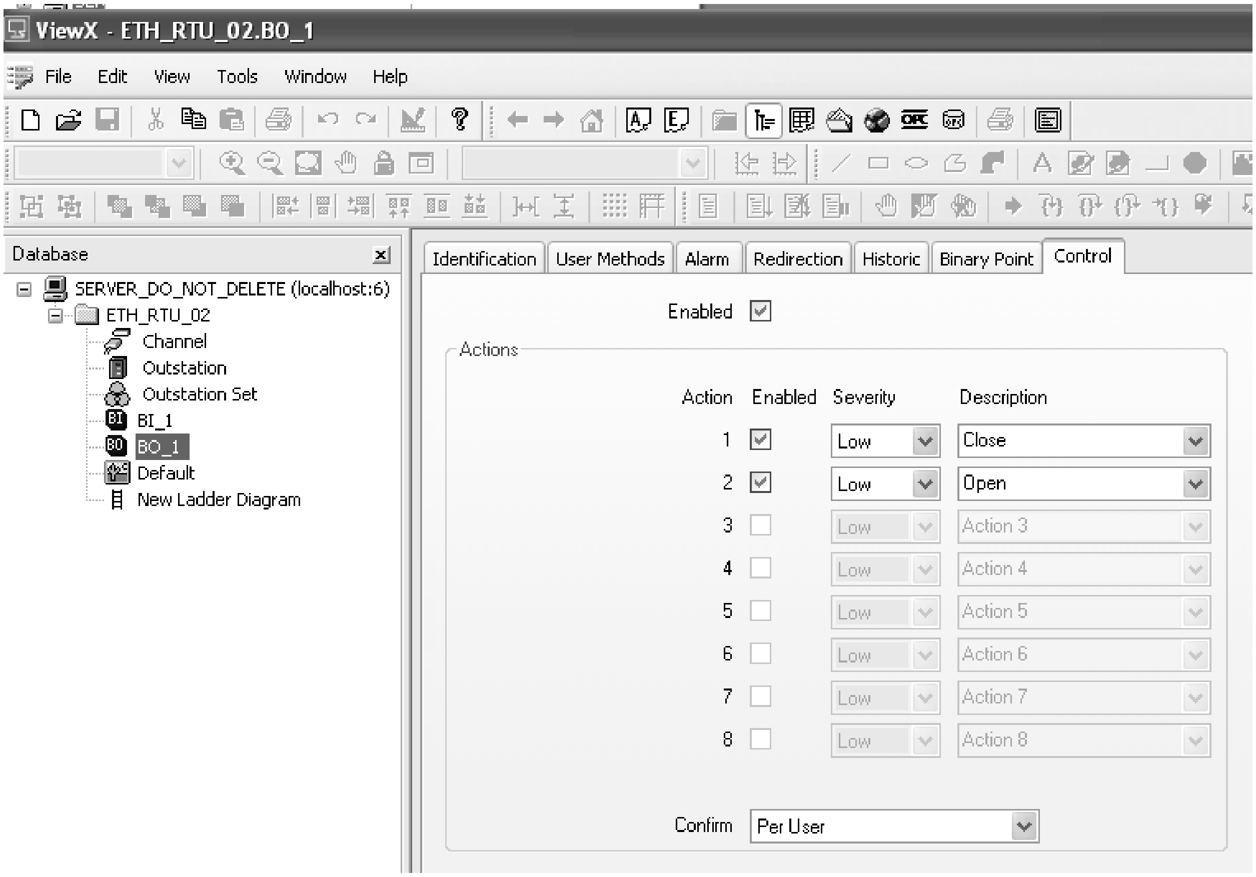

ViewX - ETH_RTU_02.BI_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Historic Binary Point
Enabled
Warning: Disabling historic will delete any historic data for this object.
Compression
Apply To No Data
Significant Change Percentage of Span 5
Minimum Interval 0 Seconds
Trend / List
Offset H-1H
Interval 2H
Relative To Latest
Use Raw Data
Stepped Mode
Total Multiplier 1

ViewX - ETH_RTU_02.BI_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Historic Binary Point
In Service
Confirm Disable/Enable Per User
Outstation ETH_RTU_02. Outstation Go To
Point Number 10
Number of Bits 1 Double Bit
Severity Low
Scan Group (Confidence Poll) None
Level 3 Scan
Format
States
State Severity Description
0 None State 0
1 None State 1
2 None State 2

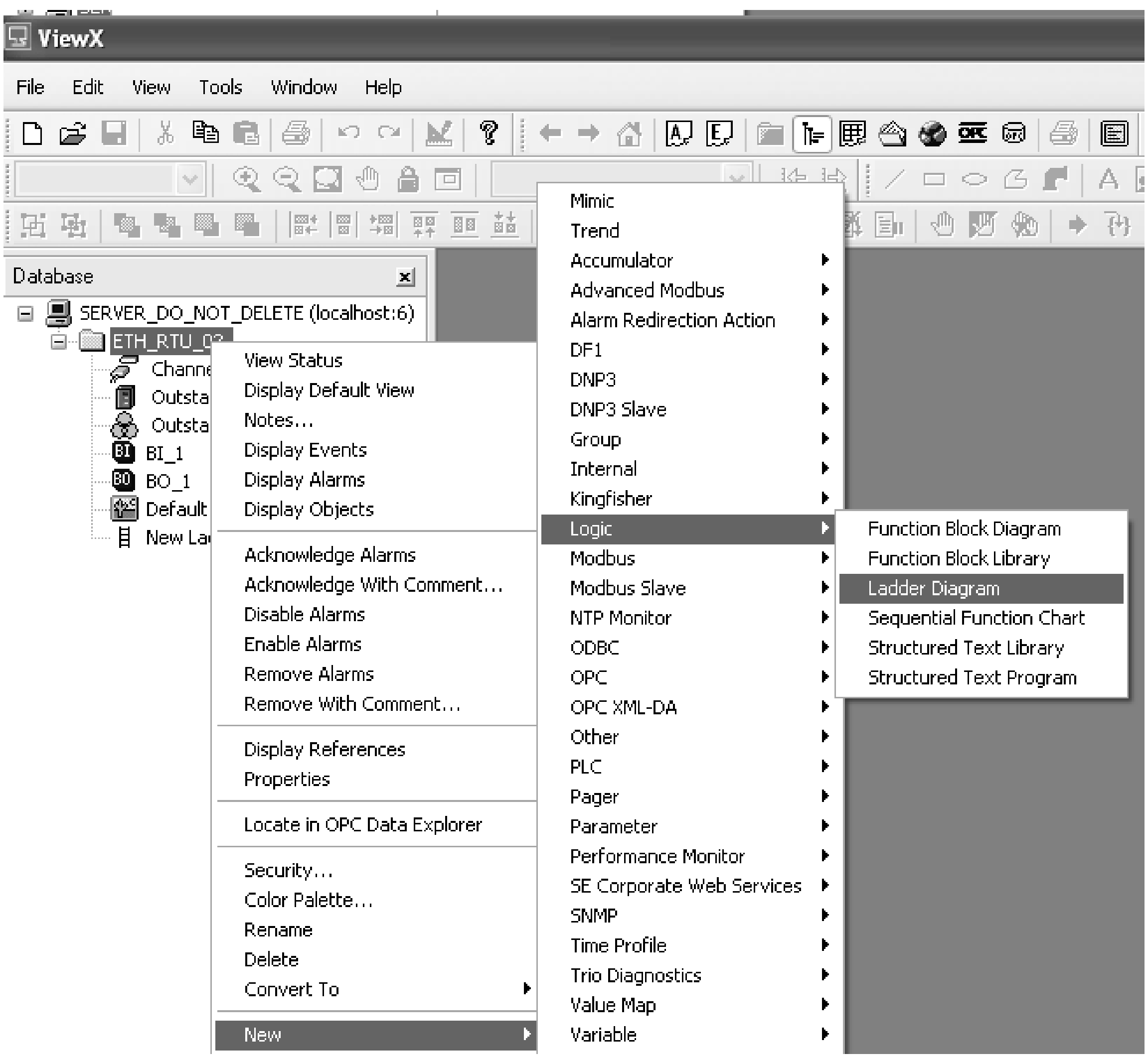

ViewX - ETH_RTU_02.BI_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification
User Methods
Alarm
Redirection
Historic
Binary Point
6 None State 6
7 None State 7
Override
Enabled
Confirm Per User
Process Override Flag
Trend
Offset Minute - 10 Minutes
Interval 15 Minutes
Update Rate 15 Seconds
Logged Data
Historic Data Filter Static and Event Data
Alarms From All Data

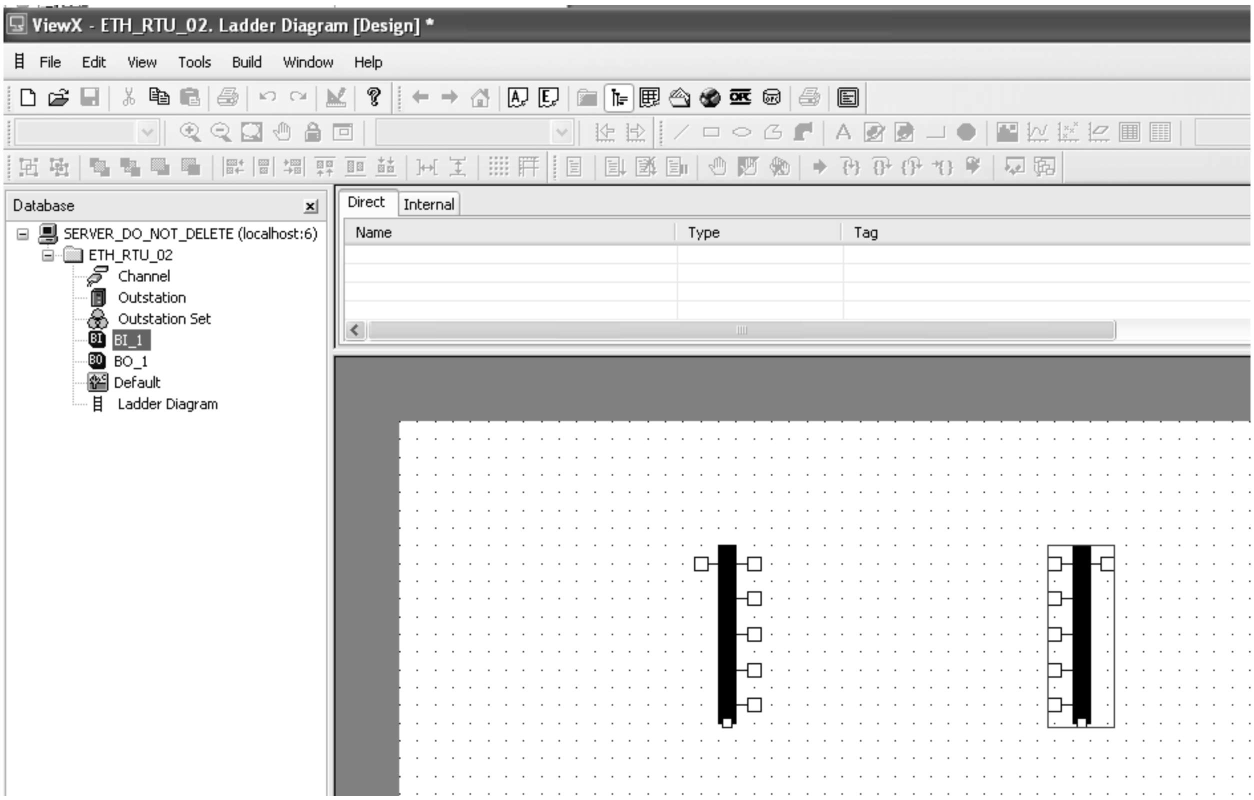

ViewX - ETH_RTU_02.BO_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification
User Methods
Alarm
Redirection
Historic
Binary Point
Control
Enabled
Warning: Disabling historic will delete any historic data for this object.
Compression
Apply To No Data
Significant Change Percentage of Span 5
Minimum Interval 0 Seconds
Trend / List
Offset H-1H
Interval 2H
Relative To Latest
Use Raw Data
Stepped Mode
Total Multiplier 1

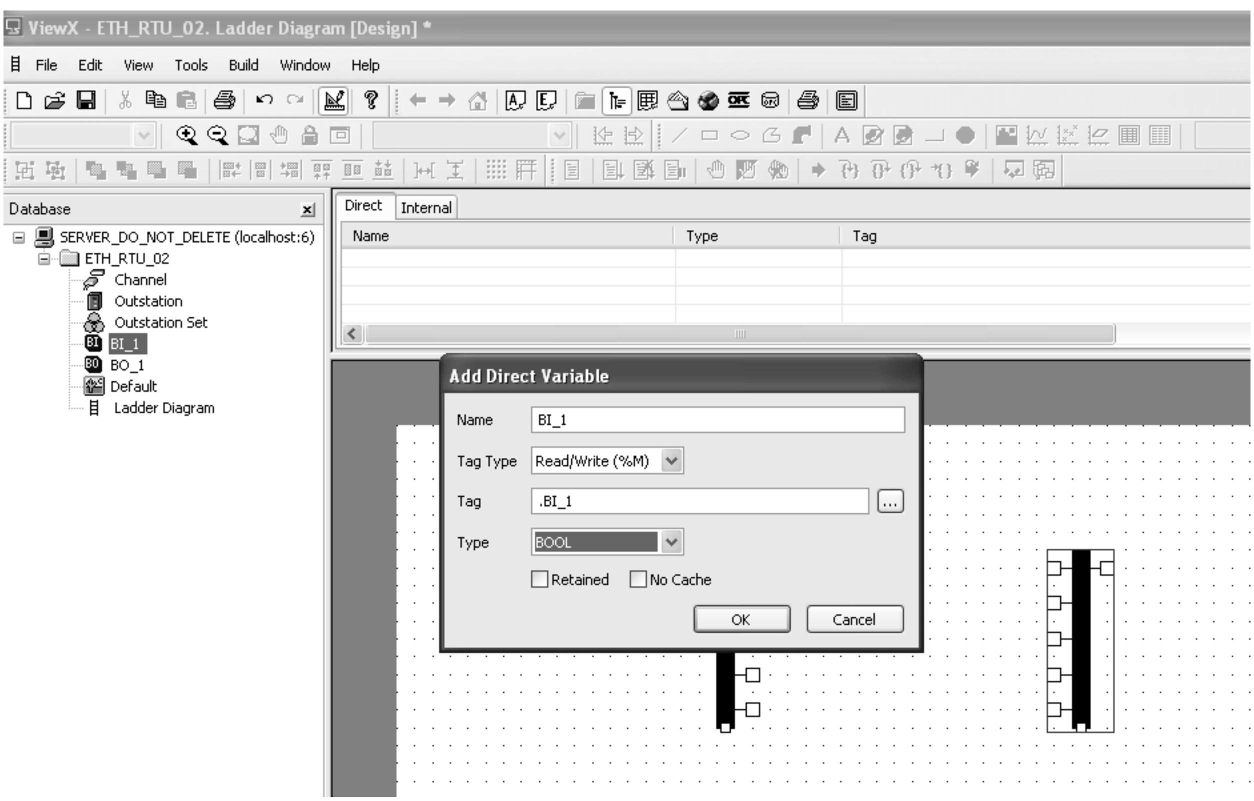

ViewX - ETH_RTU_02.BO_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Historic Binary Point Control
In Service
Confirm Disable/Enable Per User
Outstation ETH_RTU_02. Outstation Go To
Point Number 10
Number of Bits 1
Severity Low
Scan Group (Confidence Poll) None
Level 3 Scan
Format
States
State Severity Description Color
0 None State 0
1 None State 1

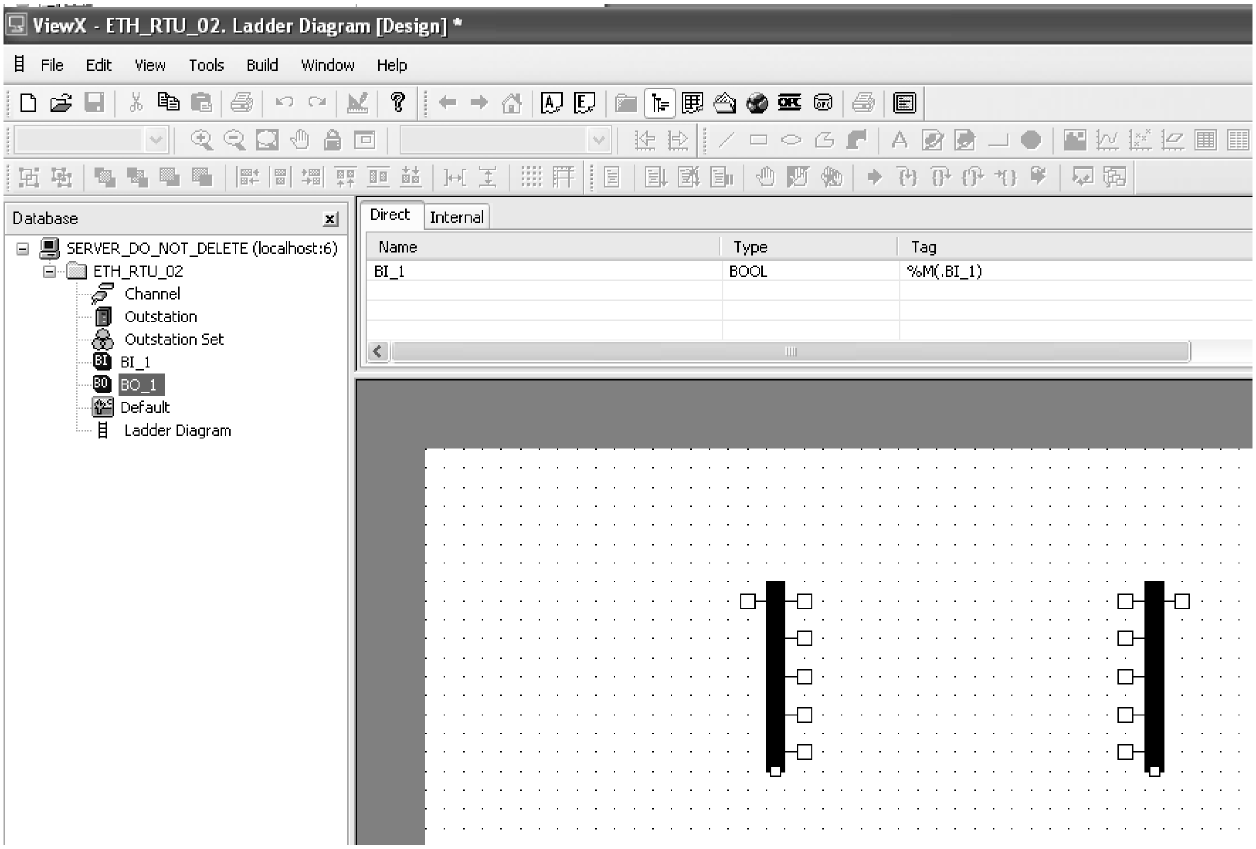

ViewX - ETH_RTU_02.BO_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification User Methods Alarm Redirection Historic Binary Point Control
6 None State 6
7 None State 7
Override
Enabled
Confirm Per User
Process Override Flag
Trend
Offset Minute - 10 Minutes
Interval 15 Minutes
Update Rate 15 Seconds
Logged Data
Historic Data Filter Static and Event Data
Alarms From All Data


ViewX - ETH_RTU_02.BO_1
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
New Ladder Diagram
Identification
User Methods
Alarm
Redirection
Historic
Binary Point
Control
Enabled
Actions
Action
Enabled
Severity
Description
1
Low
Close
2
Low
Open
3
Low
Action 3
4
Low
Action 4
5
Low
Action 5
6
Low
Action 6
7
Low
Action 7
8
Low
Action 8
Confirm
Per User

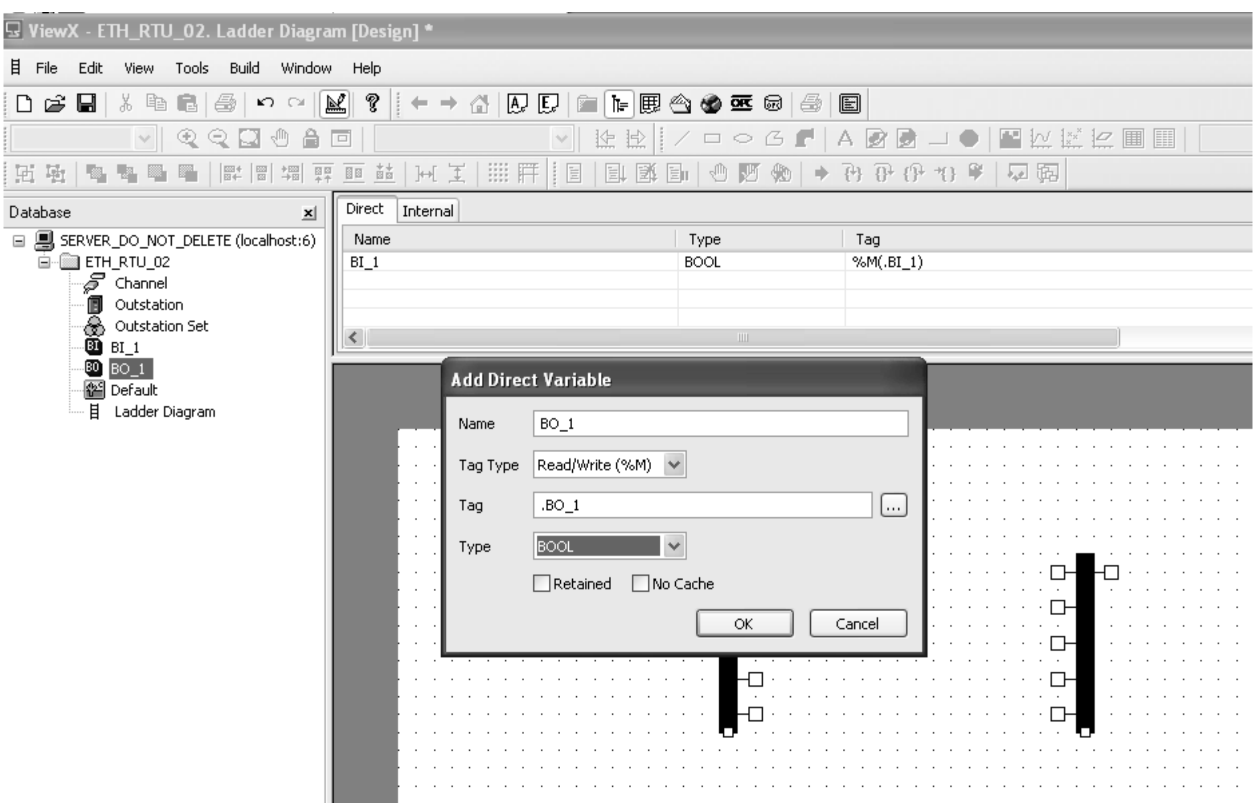

ViewX
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Function Block Diagram
Function Block Library
Ladder Diagram
Sequential Function Chart
Structured Text Library
Structured Text Program


ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct
Internal
Name
Type
Tag

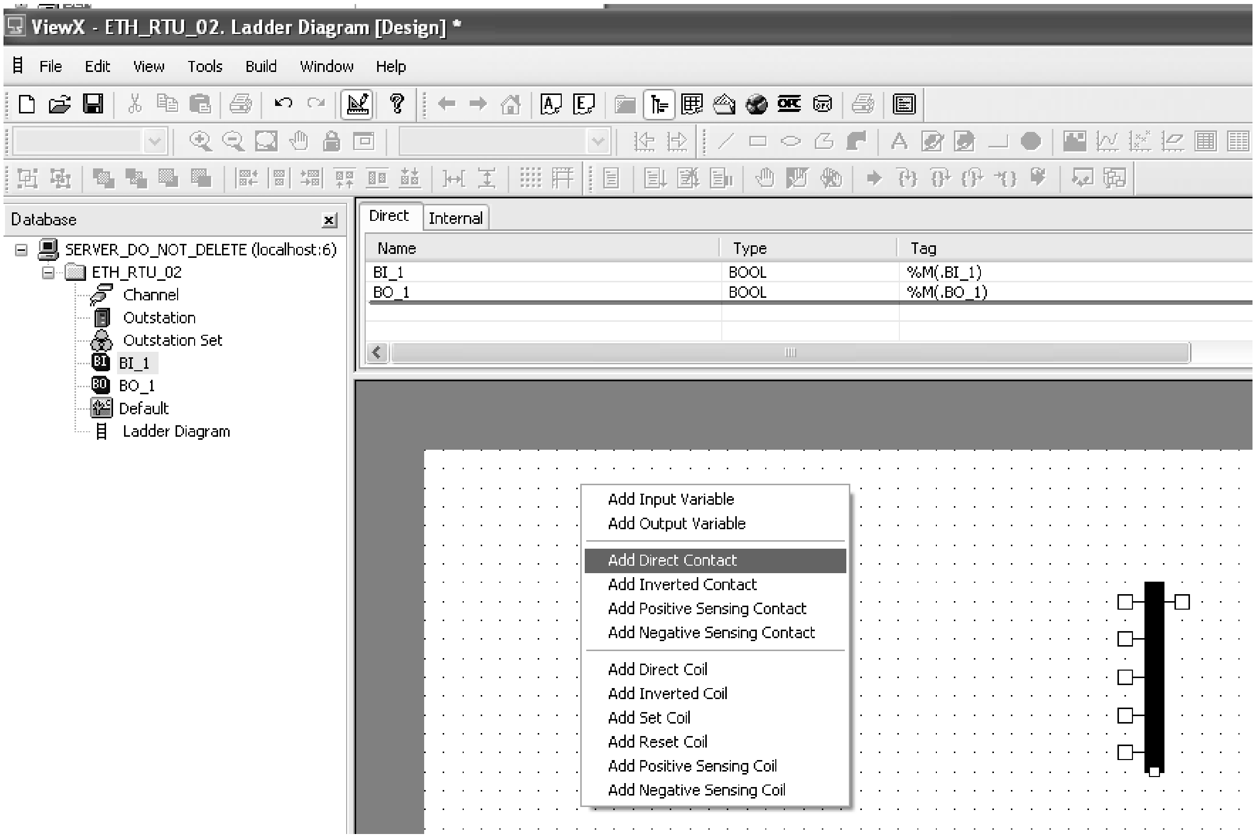

ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct
Internal
Name
Type
Tag
Add Direct Variable
Name
BI_1
Tag Type
Read/Write (%M)
Tag
.BI_1
Type
BOOL
Retained
No Cache
OK
Cancel

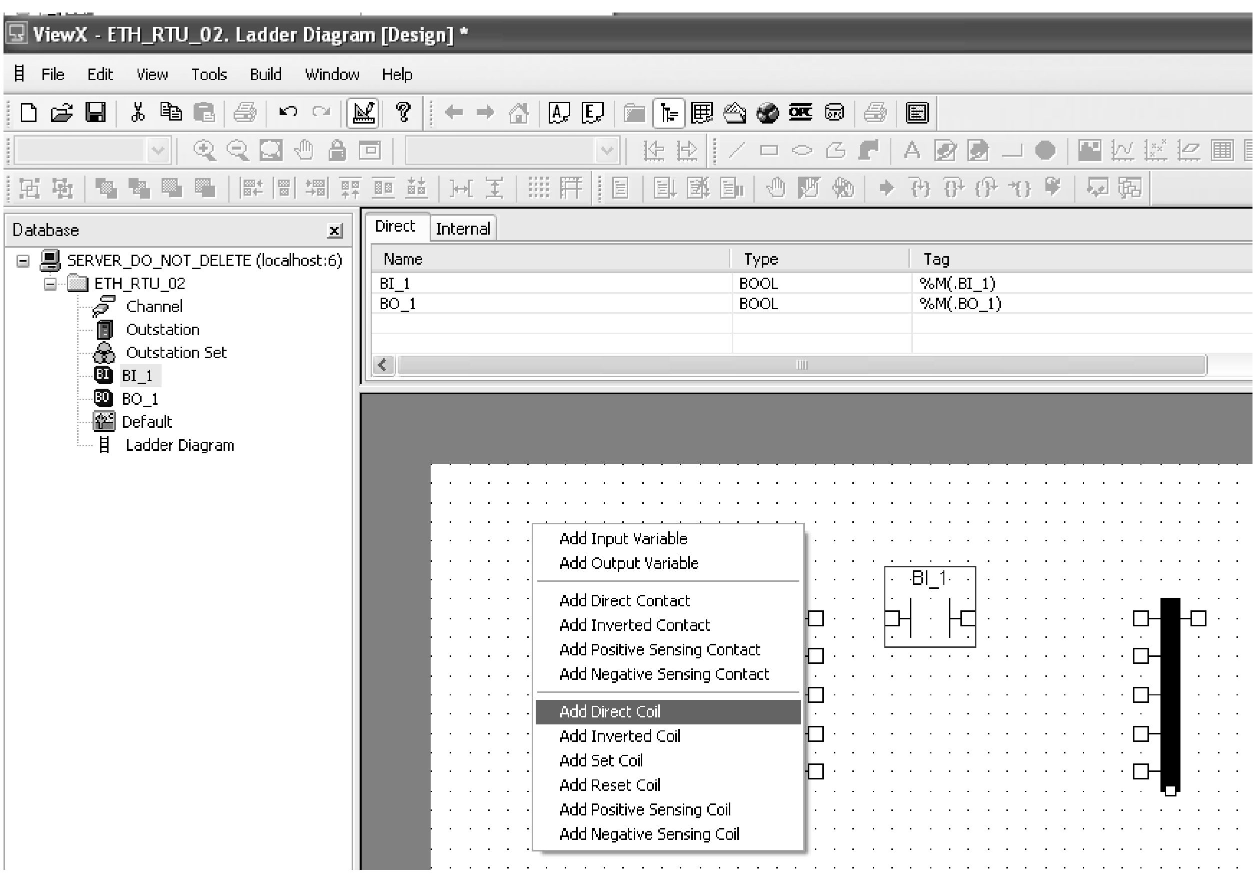


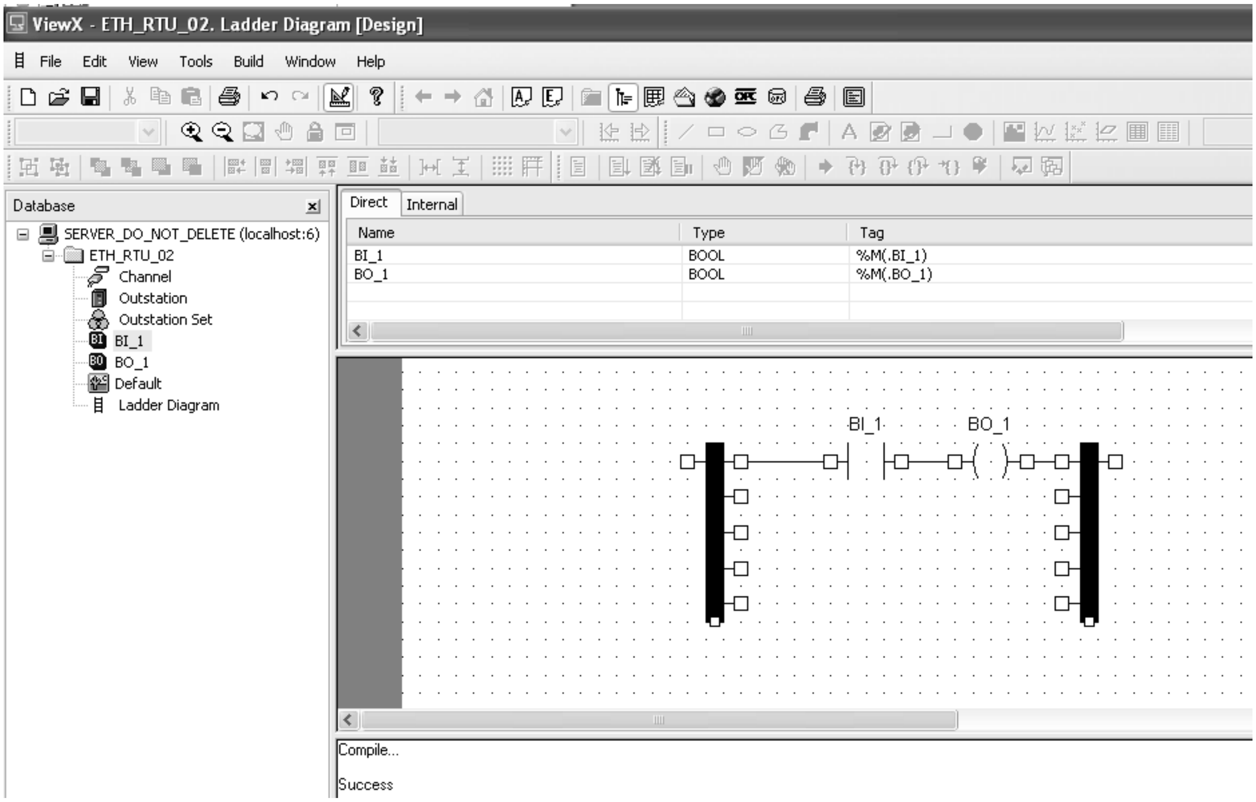
ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct Internal
Name Type Tag
BI_1 BOOL %M(.BI_1)

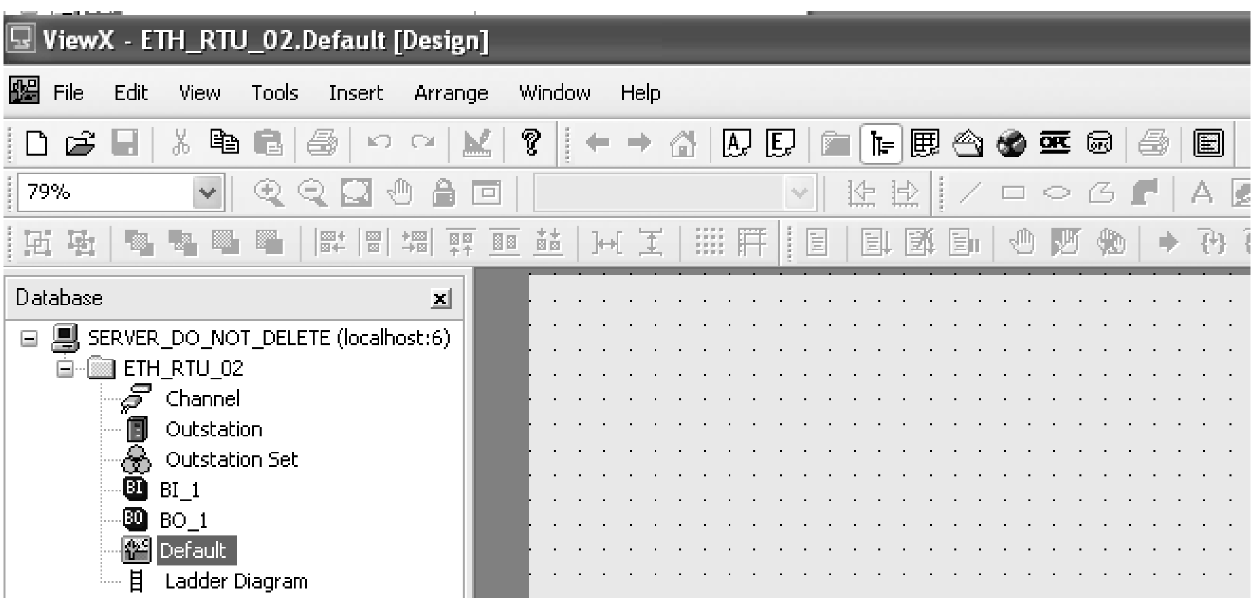


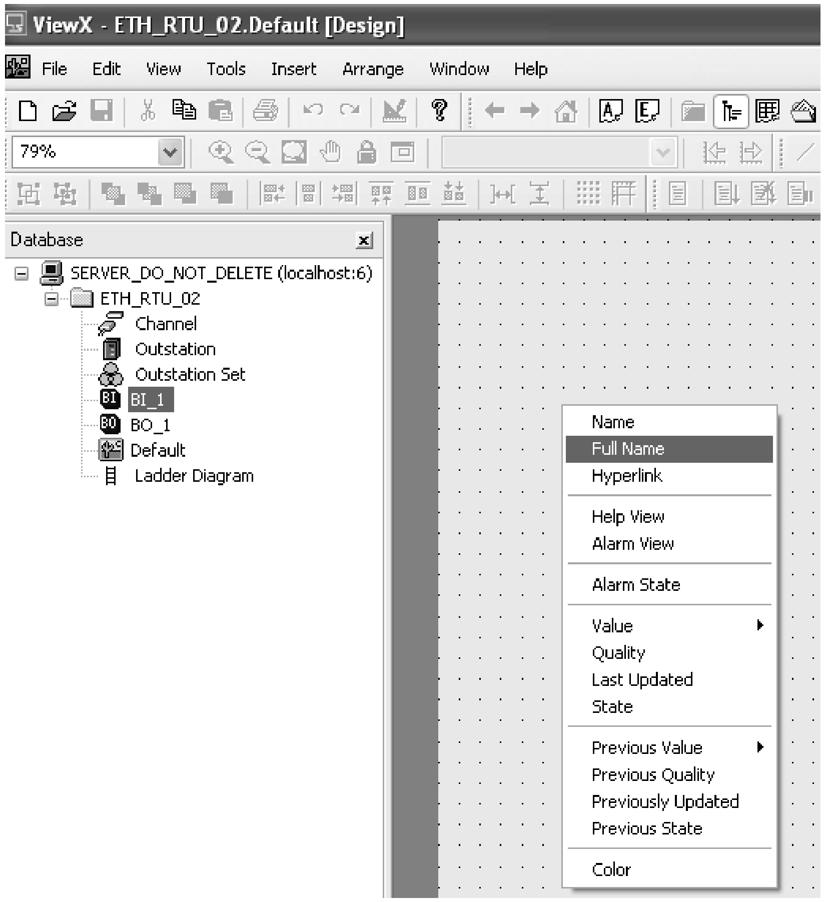
ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct Internal
Name Type Tag
BI_1 BOOL %M(.BI_1)
Add Direct Variable
Name BO_1
Tag Type Read/Write (%M)
Tag .BO_1
Type BOOL
Retained
No Cache
OK
Cancel

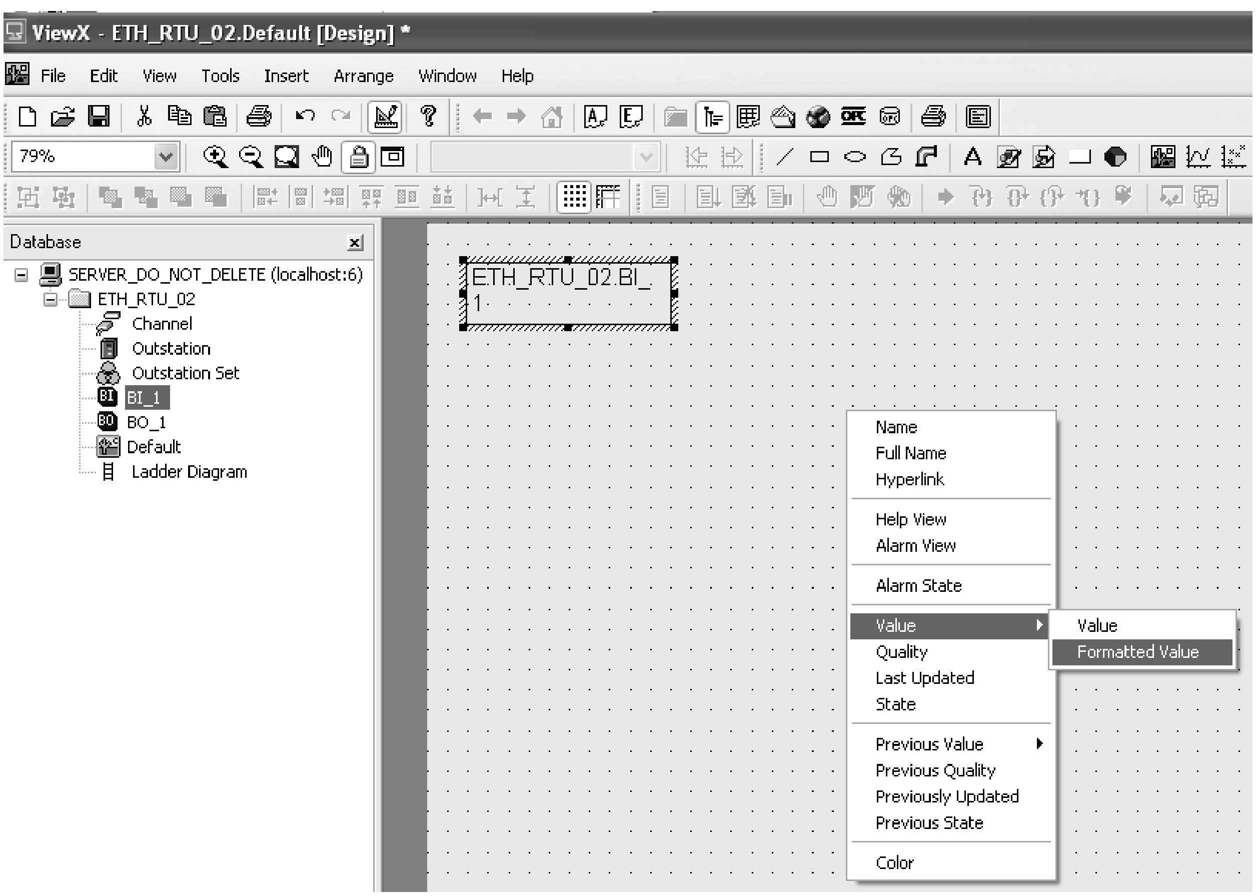

ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct Internal
Name Type Tag
BI_1 BOOL %M(.BI_1)
BO_1 BOOL %M(.BO_1)
Add Input Variable
Add Output Variable
Add Direct Contact
Add Inverted Contact
Add Positive Sensing Contact
Add Negative Sensing Contact
Add Direct Coil
Add Inverted Coil
Add Set Coil
Add Reset Coil
Add Positive Sensing Coil
Add Negative Sensing Coil

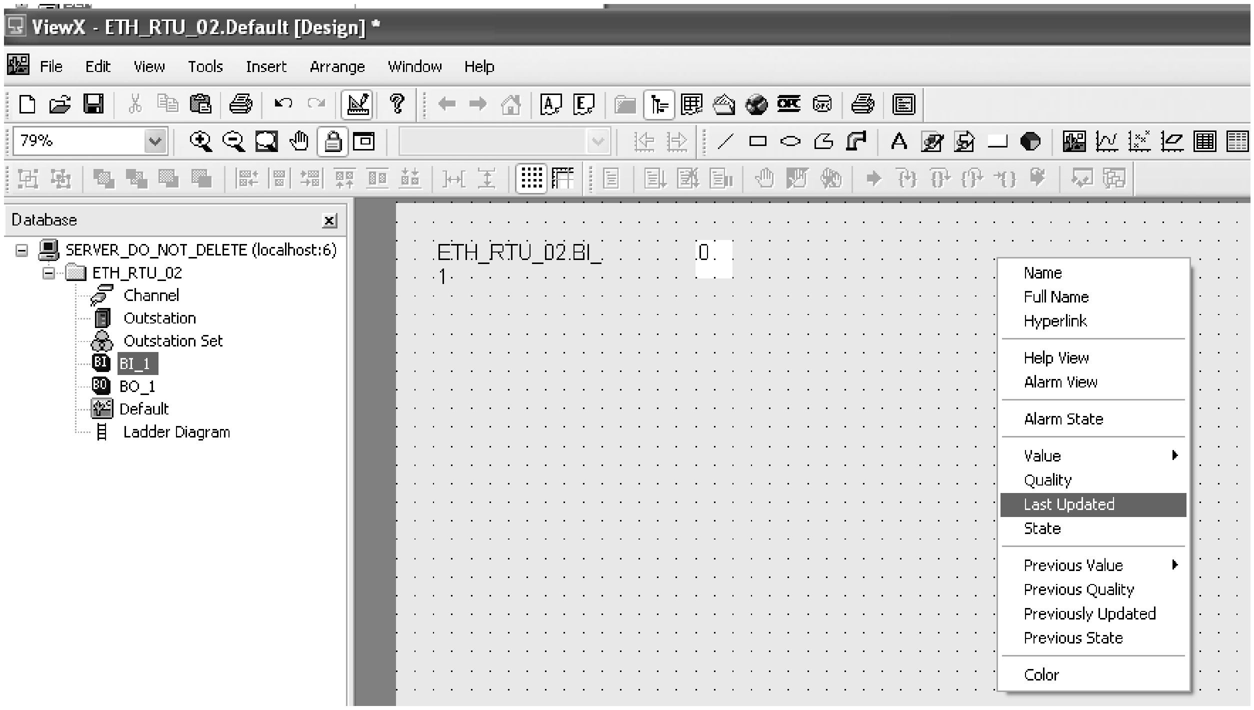

ViewX - ETH_RTU_02. Ladder Diagram [Design] *
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct Internal
Name Type Tag
BI_1 BOOL %M(.BI_1)
BO_1 BOOL %M(.BO_1)
Add Input Variable
Add Output Variable
Add Direct Contact
Add Inverted Contact
Add Positive Sensing Contact
Add Negative Sensing Contact
Add Direct Coil
Add Inverted Coil
Add Set Coil
Add Reset Coil
Add Positive Sensing Coil
Add Negative Sensing Coil
BI_1

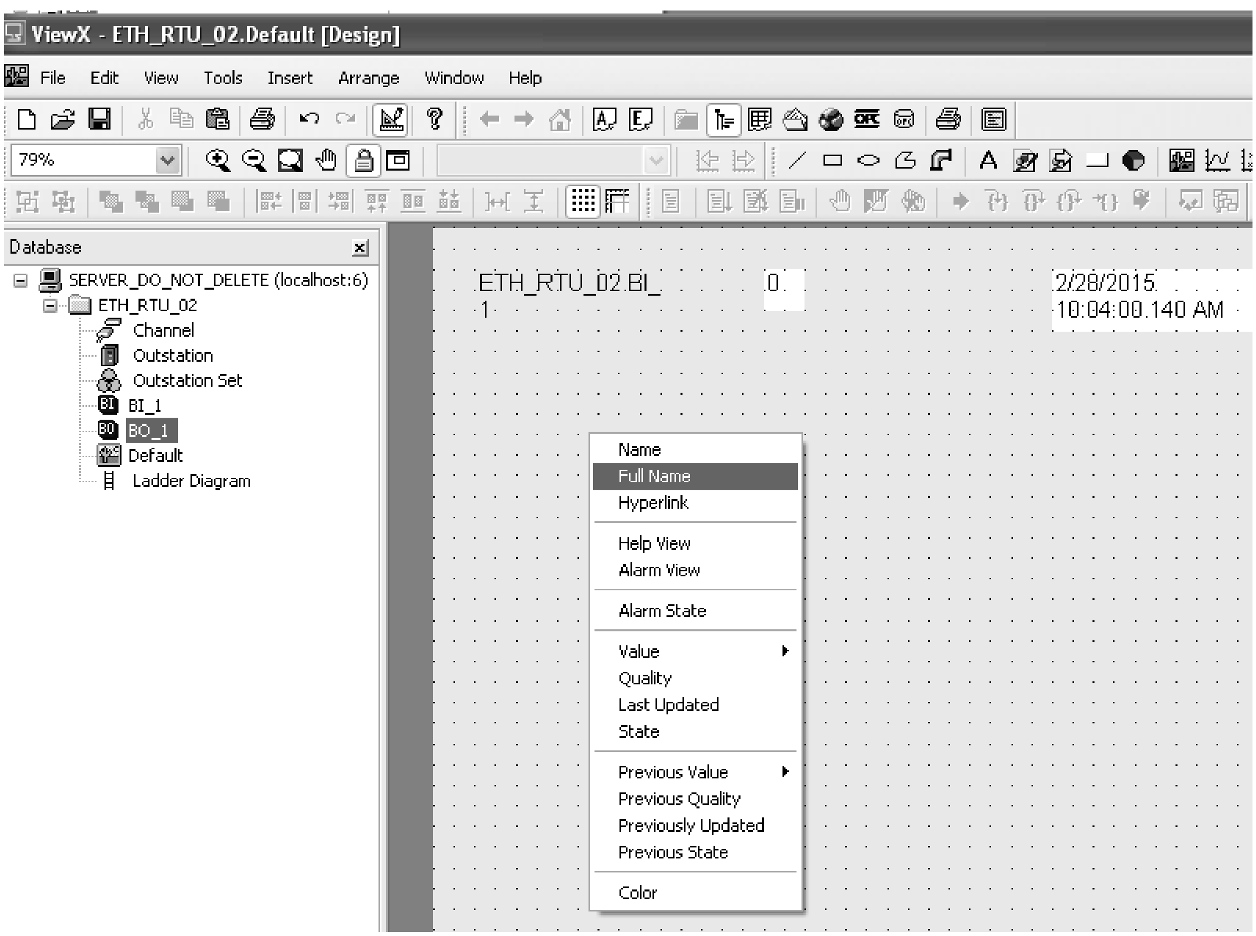

ViewX - ETH_RTU_02. Ladder Diagram [Design]
File Edit View Tools Build Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Direct Internal
Name Type Tag
BI_1 BOOL %M(.BI_1)
BO_1 BOOL %M(.BO_1)
BI_1
BO_1
Compile...
Success

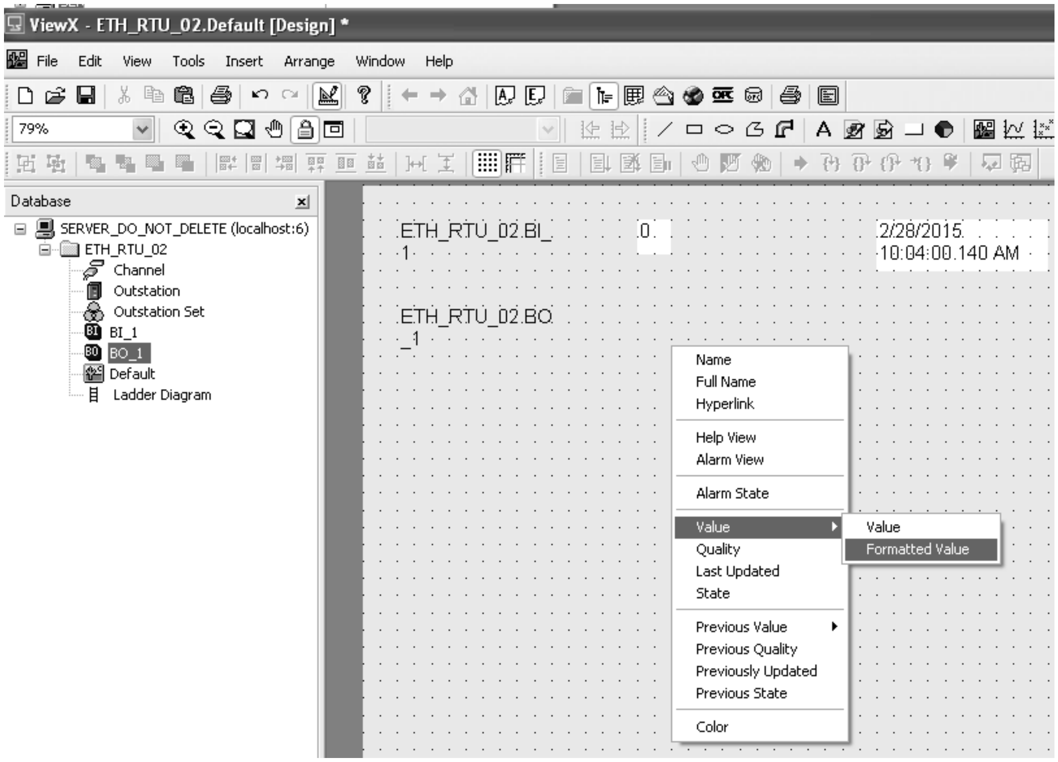

ViewX - ETH_RTU_02.Default [Design]
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram

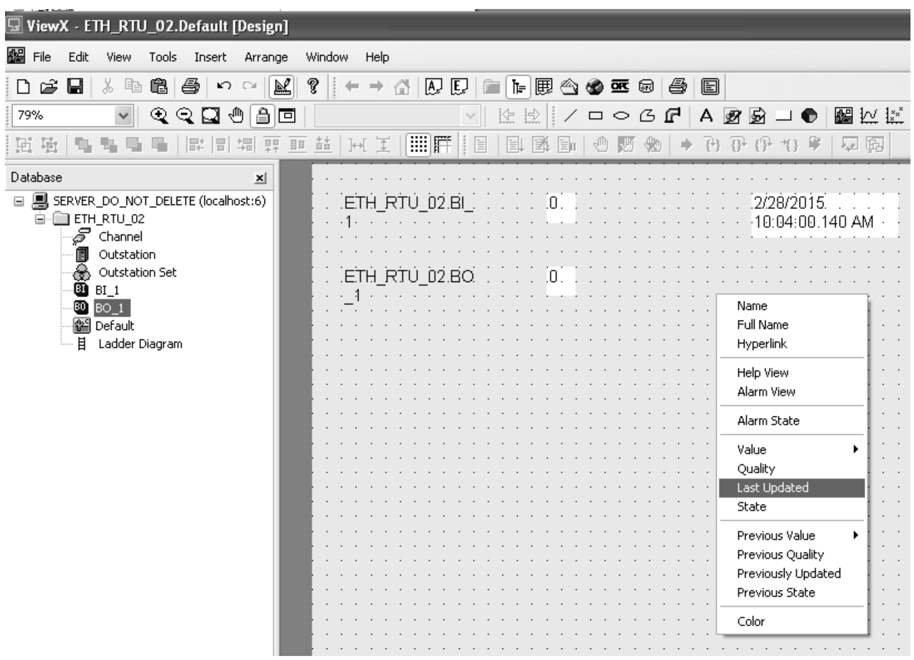

ViewX - ETH_RTU_02.Default [Design]
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
Name
Full Name
Hyperlink
Help View
Alarm View
Alarm State
Value
Quality
Last Updated
State
Previous Value
Previous Quality
Previously Updated
Previous State
Color

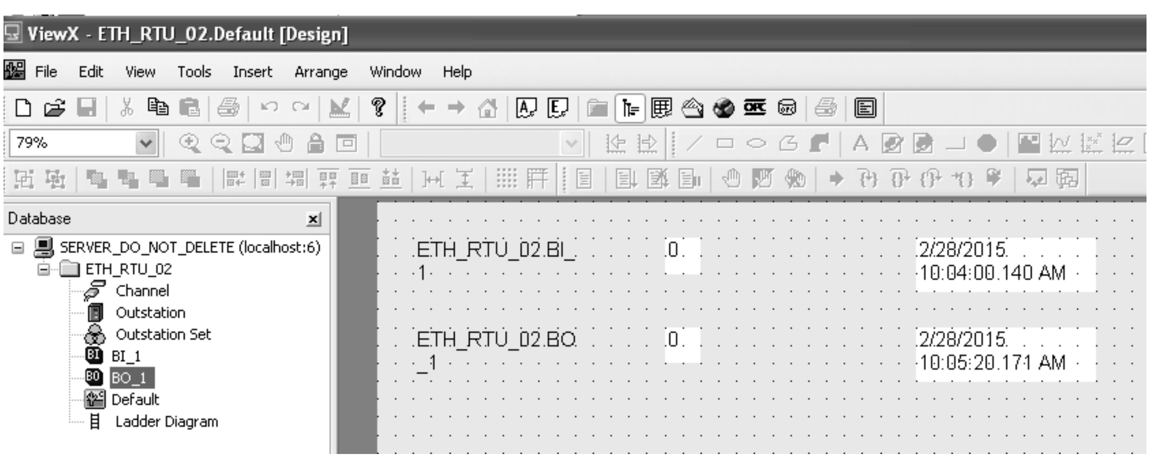

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
ETH_RTU_02.BI_1
Name
Full Name
Hyperlink
Help View
Alarm View
Alarm State
Value
Quality
Last Updated
State
Previous Value
Previous Quality
Previously Updated
Previous State
Color
Value
Formatted Value

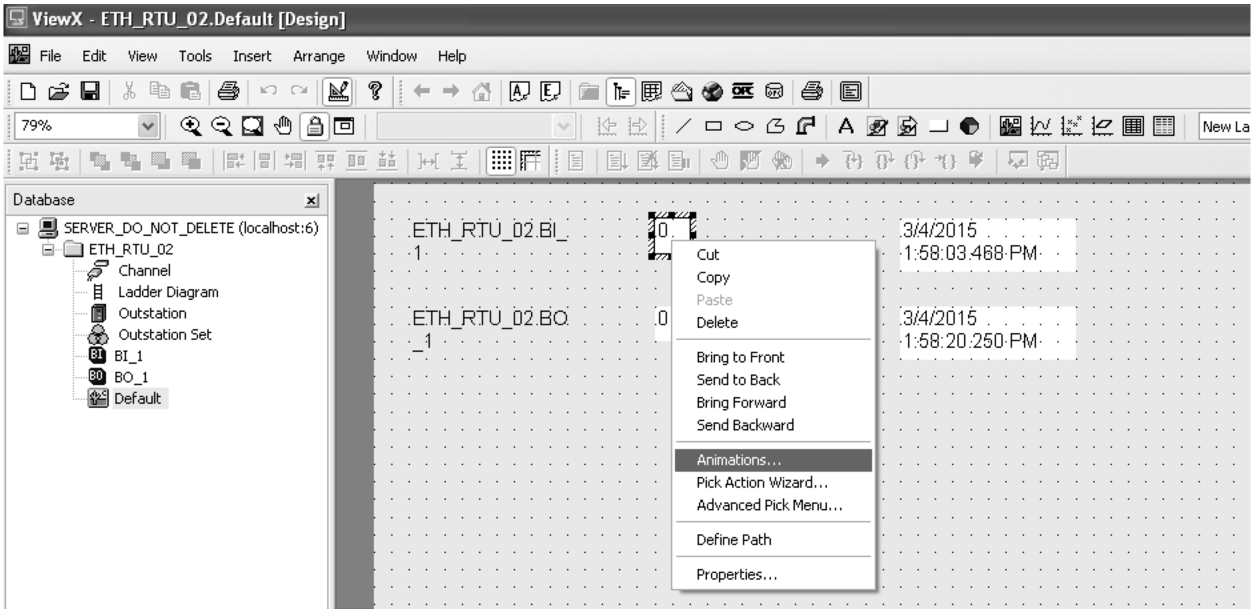


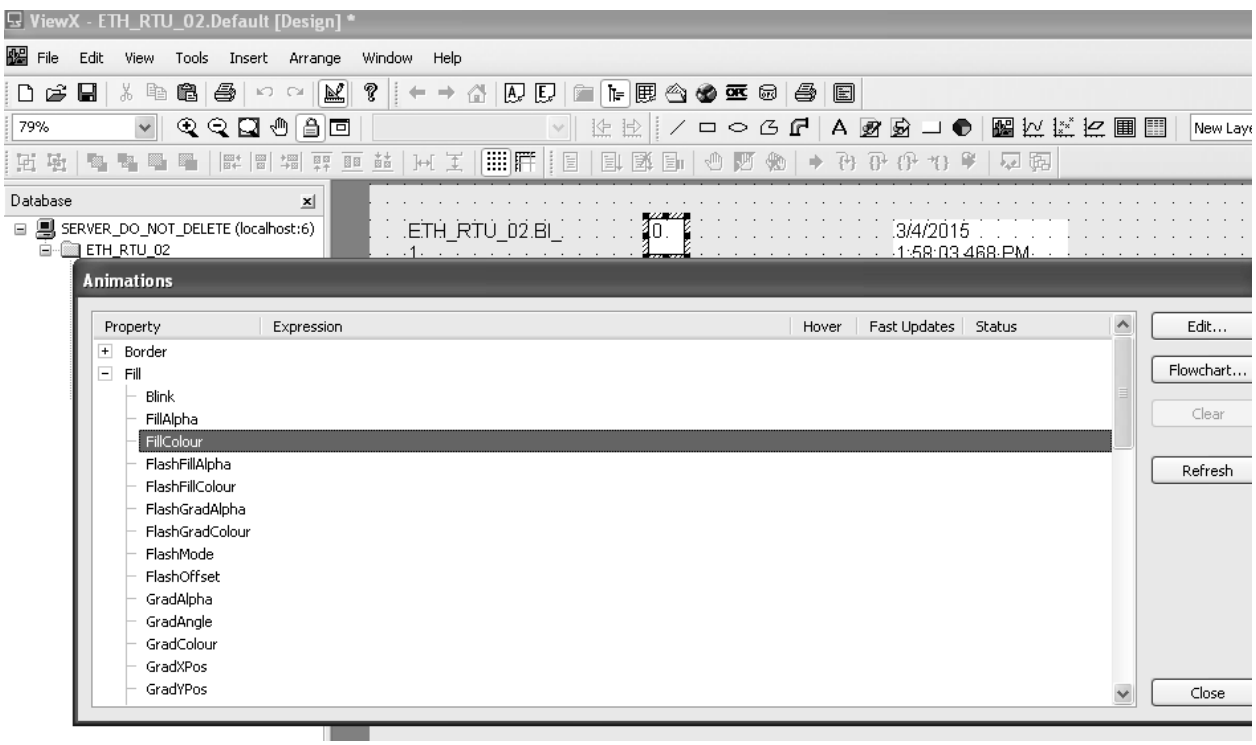


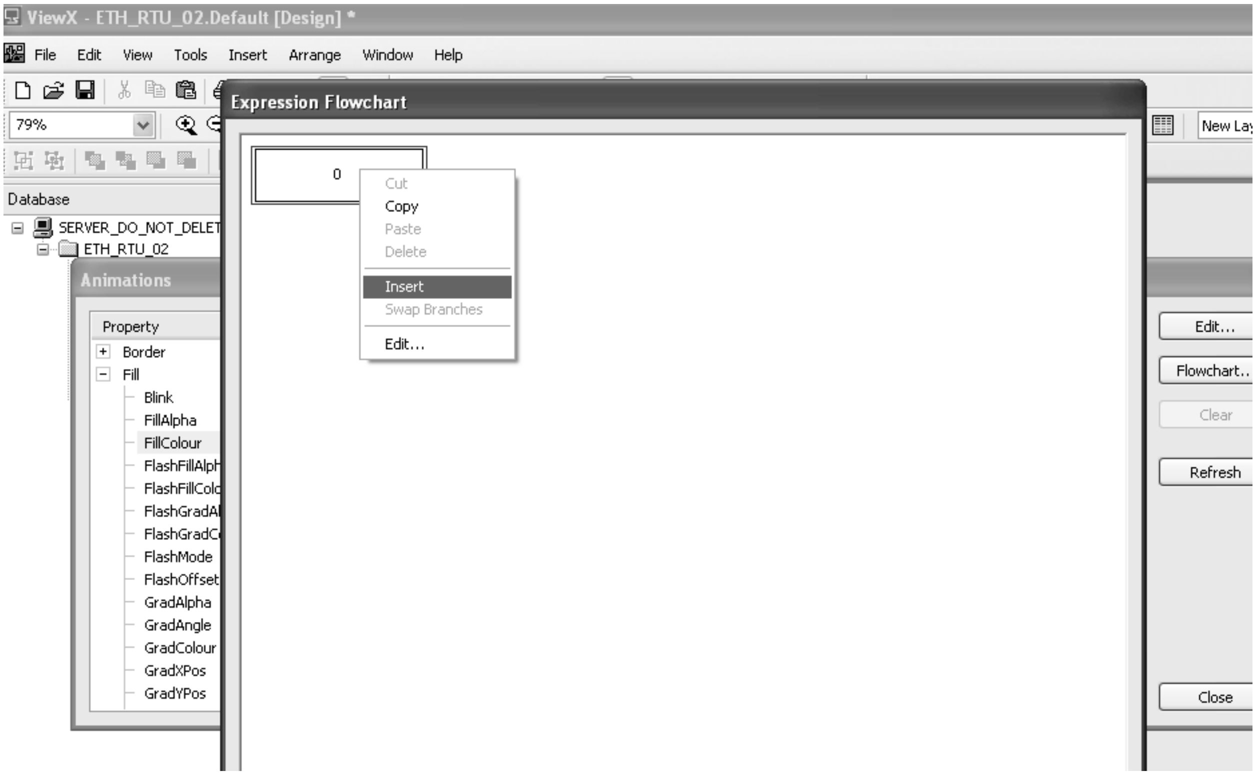

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
ETH_RTU_02.BI_1
0
2/28/2015
10:04:00.140 AM
ETH_RTU_02.BO_1
Name
Full Name
Hyperlink
Help View
Alarm View
Alarm State
Value
Quality
Last Updated
State
Previous Value
Previous Quality
Previously Updated
Previous State
Color
Value
Formatted Value

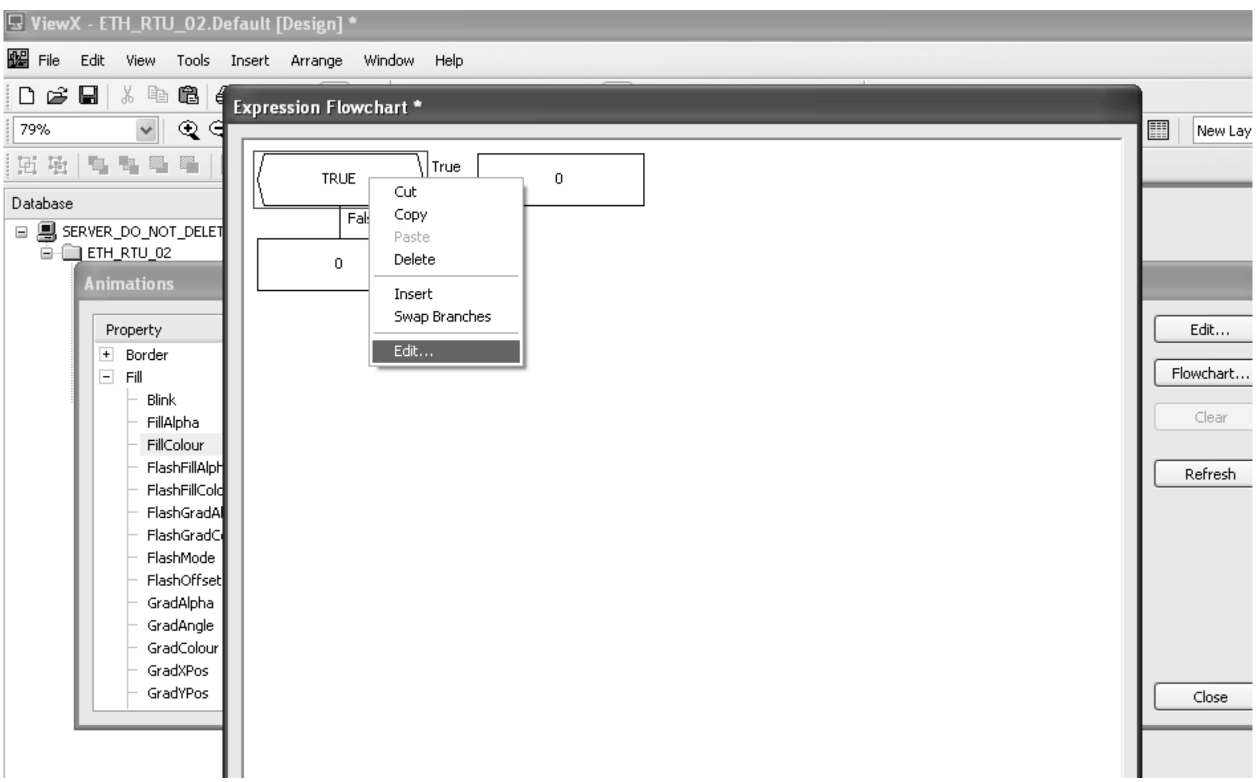

ViewX - ETH_RTU_02.Default [Design]
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
ETH_RTU_02.BI_1
0
2/28/2015
10:04:00.140 AM
ETH_RTU_02.BO_1
0
Name
Full Name
Hyperlink
Help View
Alarm View
Alarm State
Value
Quality
Last Updated
State
Previous Value
Previous Quality
Previously Updated
Previous State
Color

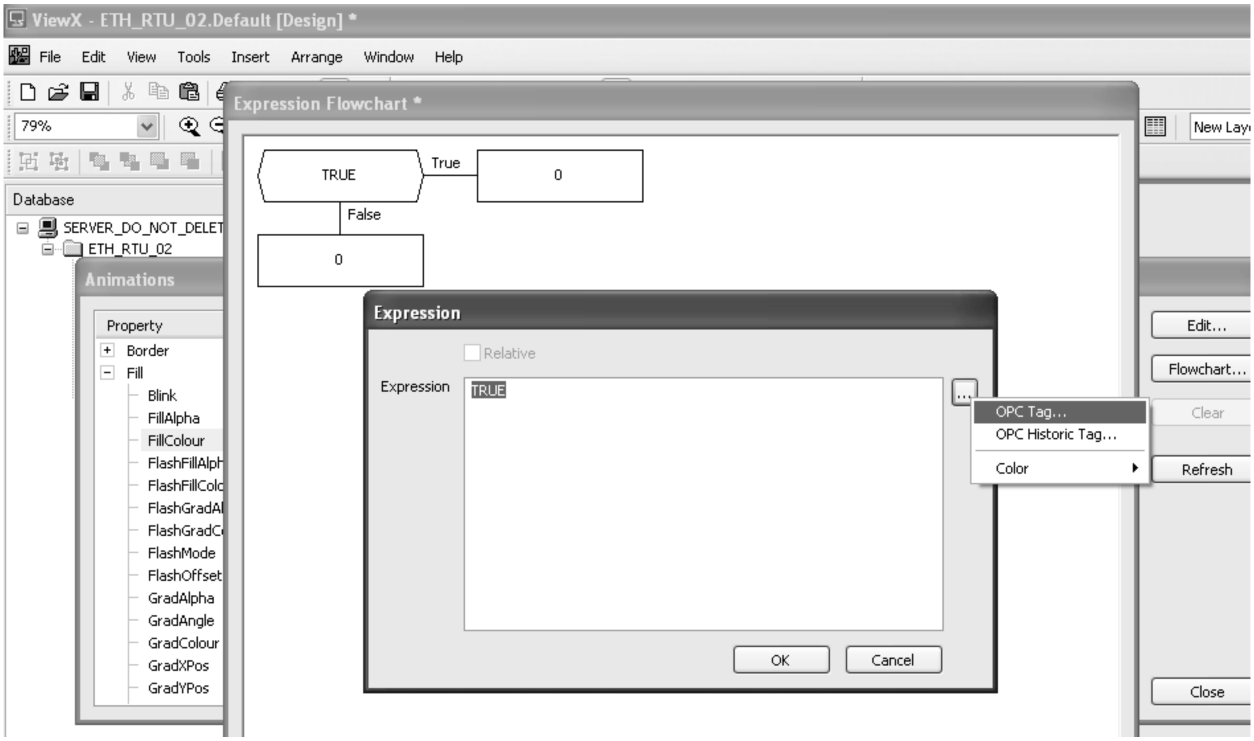

ViewX - ETH_RTU_02.Default [Design]
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Outstation
Outstation Set
BI_1
BO_1
Default
Ladder Diagram
ETH_RTU_02.BI_1
0
2/28/2015
10:04:00.140 AM
ETH_RTU_02.BO_1
0
2/28/2015
10:05:20.171 AM

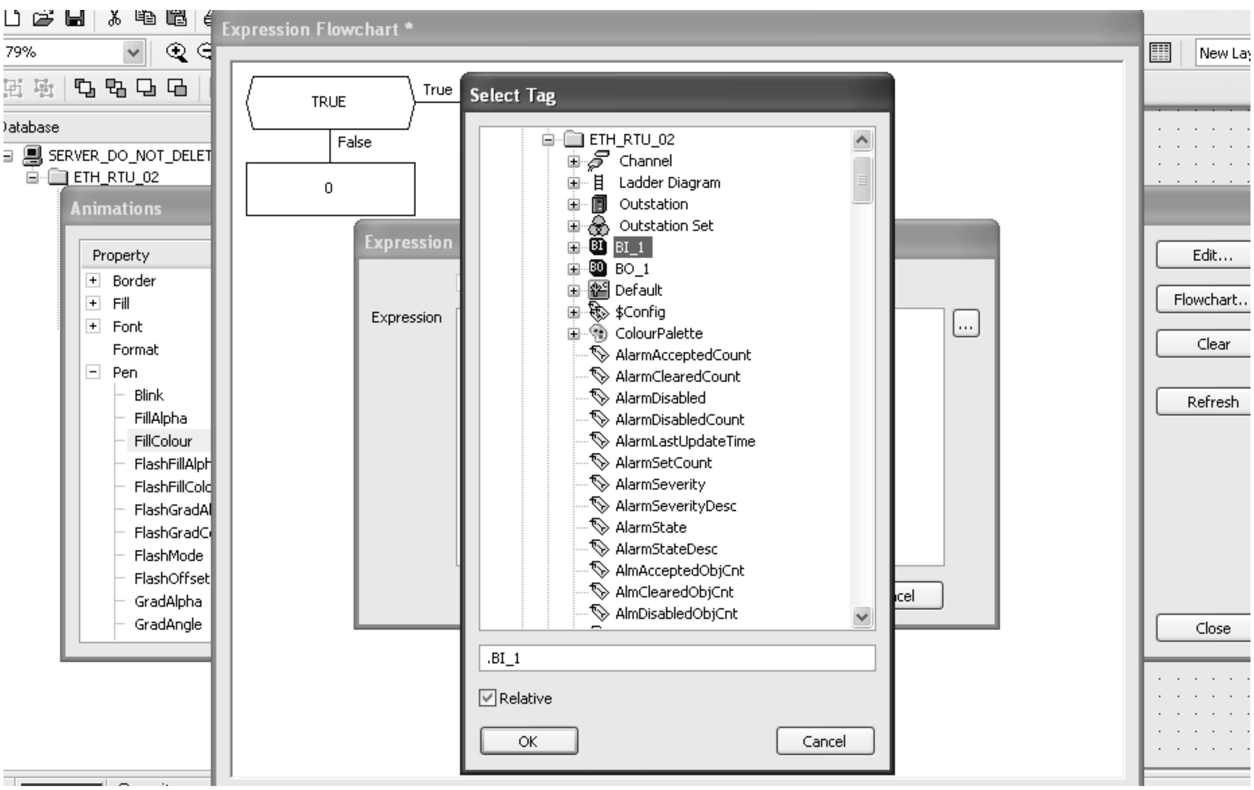

ViewX - ETH_RTU_02.Default [Design]
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
ETH_RTU_02.BI_1
ETH_RTU_02.BO_1
3/4/2015 1:58:03.468 PM
3/4/2015 1:58:20.250 PM
Cut
Copy
Paste
Delete
Bring to Front
Send to Back
Bring Forward
Send Backward
Animations...
Pick Action Wizard...
Advanced Pick Menu...
Define Path
Properties...

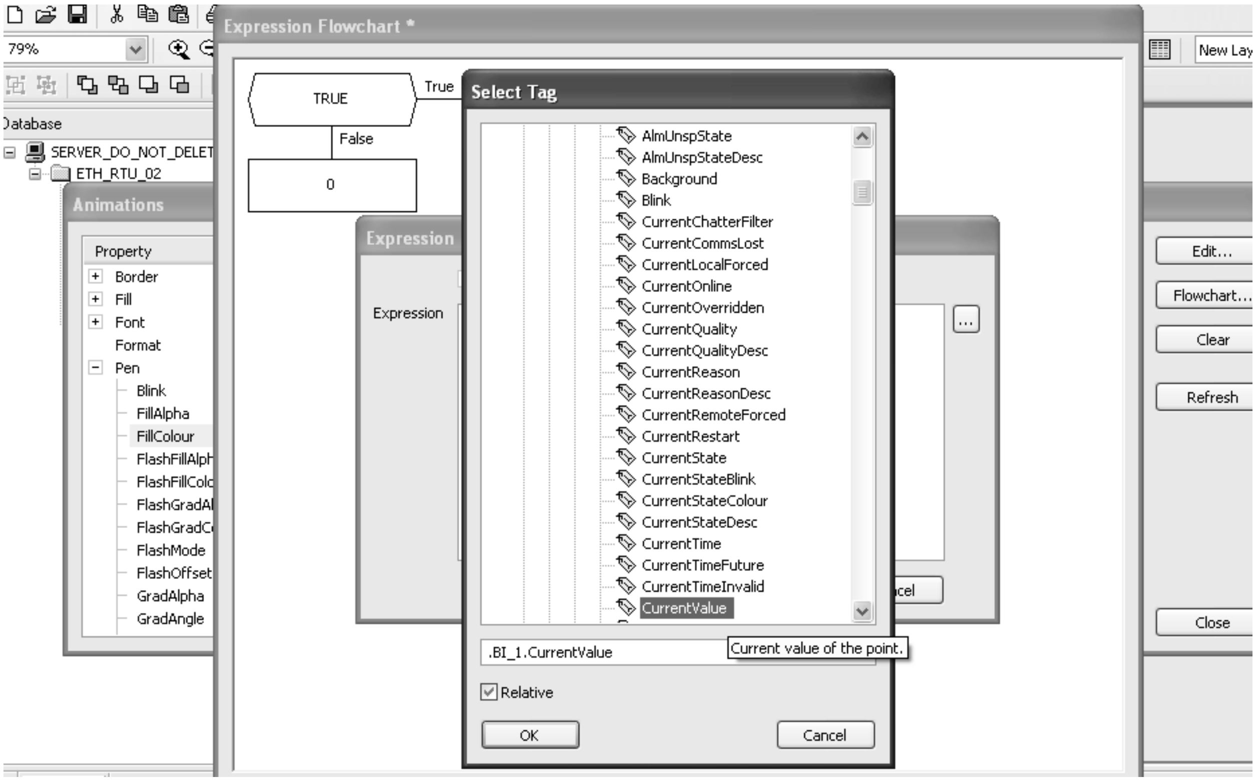

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Animations
Property
Expression
Hover
Fast Updates
Status
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlpha
FlashFillColour
FlashGradAlpha
FlashGradColour
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

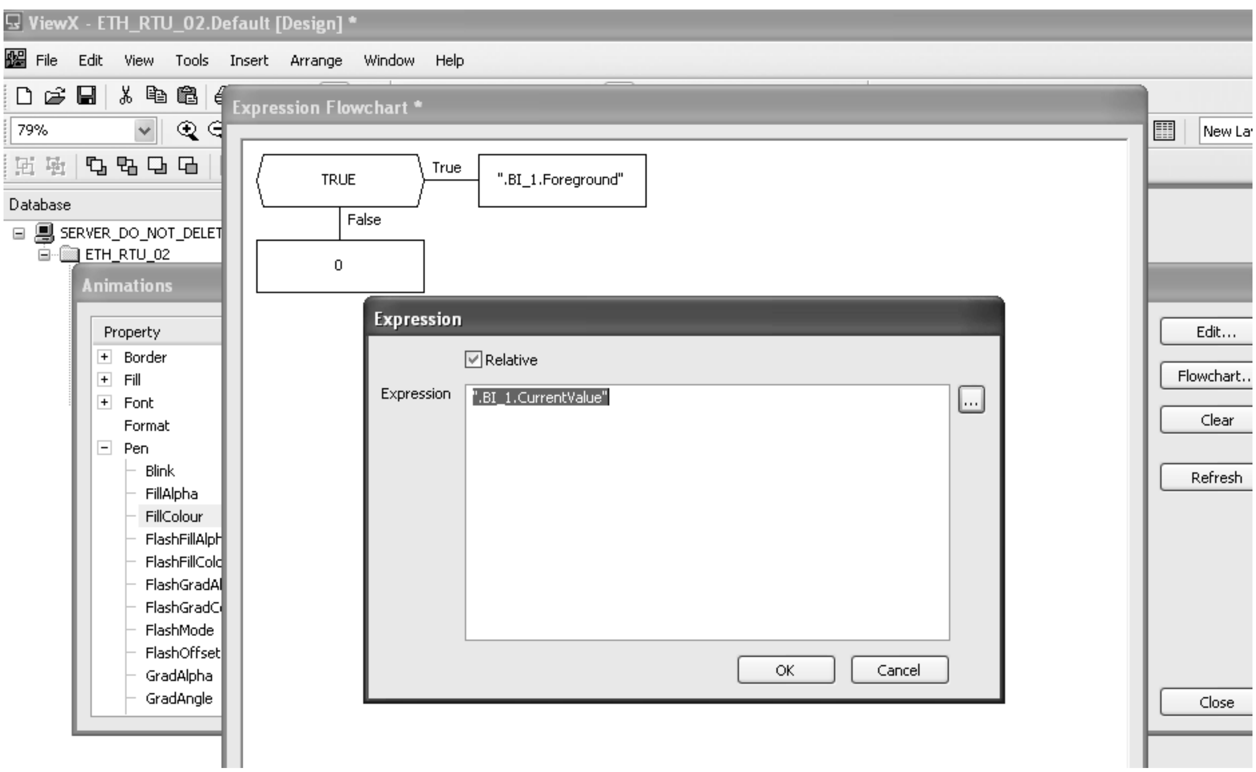

ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
Expression Flowchart
0
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

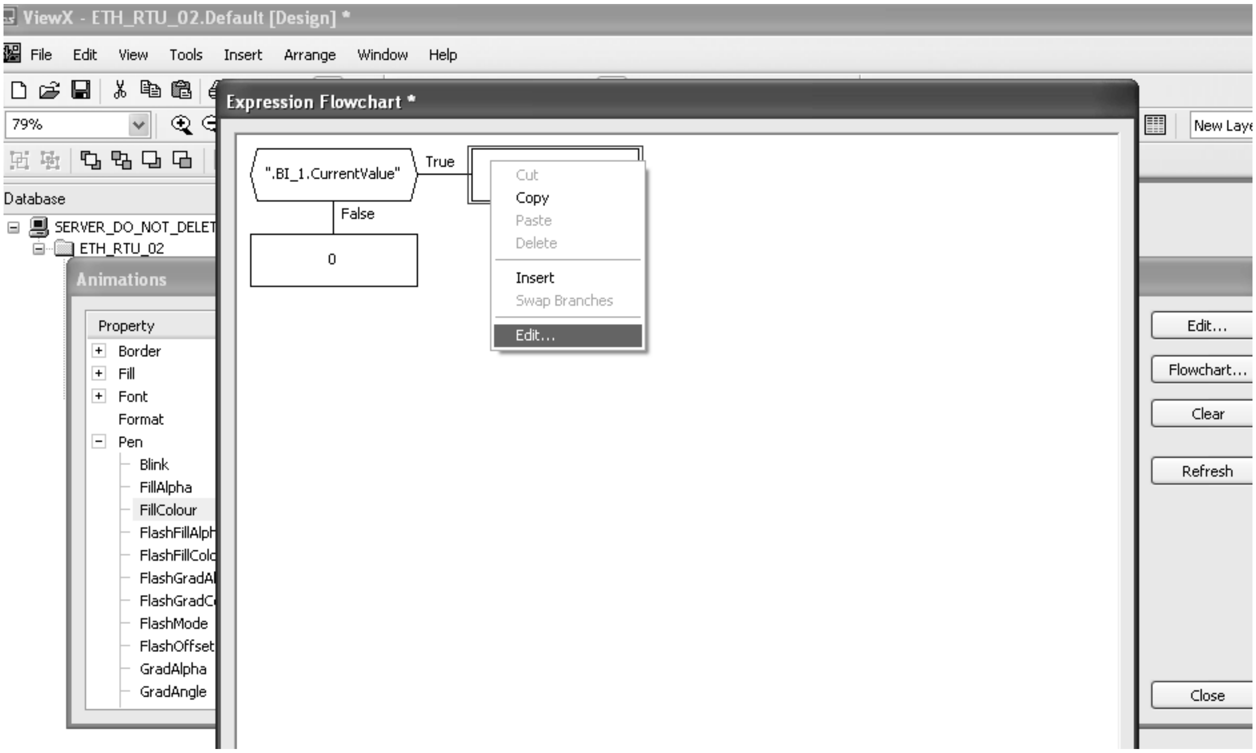

ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
Expression Flowchart *
TRUE
True
0
0
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close


ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
TRUE
True
0
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Expression
Relative
Expression
TRUE
OPC Tag...
OPC Historic Tag...
Color
OK
Cancel
Edit...
Flowchart...
Clear
Refresh
Close

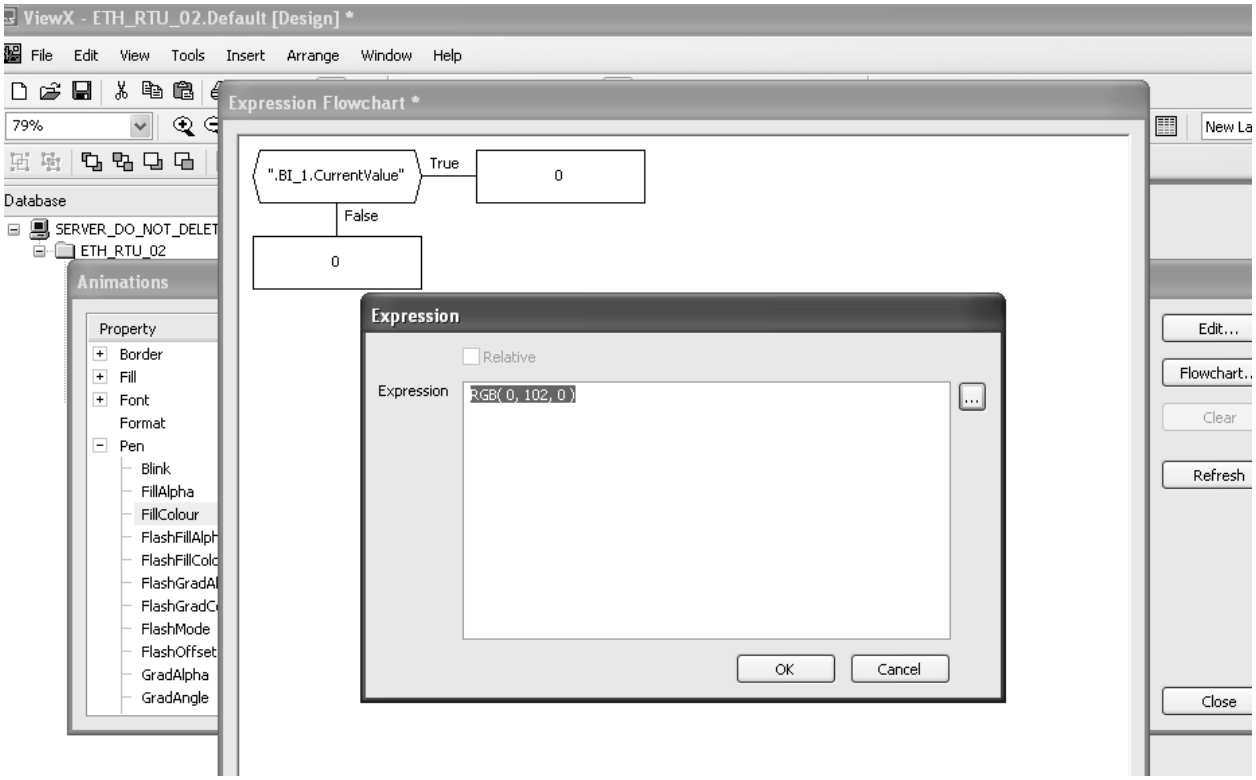

Expression Flowchart *
TRUE
True
False
0
Select Tag
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
$Config
ColourPalette
AlarmAcceptedCount
AlarmClearedCount
AlarmDisabled
AlarmDisabledCount
AlarmLastUpdateTime
AlarmSetCount
AlarmSeverity
AlarmSeverityDesc
AlarmState
AlarmStateDesc
AlmAcceptedObjCnt
AlmClearedObjCnt
AlmDisabledObjCnt
.BI_1
Relative
OK
Cancel
Property
Border
Fill
Font
Format
Pen
Edit...
Flowchart...
Clear
Refresh
Close

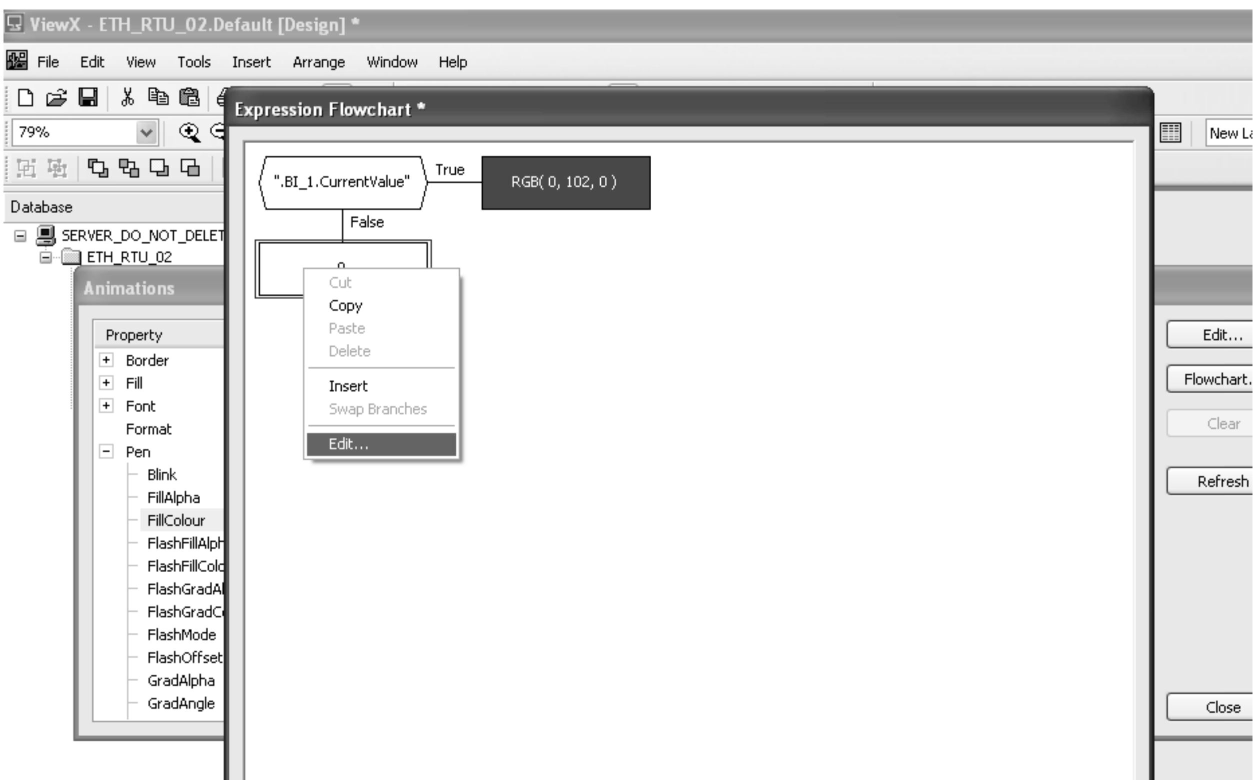

Expression Flowchart *
TRUE
True
False
0
Select Tag
AlmUnspState
AlmUnspStateDesc
Background
Blink
CurrentChatterFilter
CurrentCommsLost
CurrentLocalForced
CurrentOnline
CurrentOverridden
CurrentQuality
CurrentQualityDesc
CurrentReason
CurrentReasonDesc
CurrentRemoteForced
CurrentRestart
CurrentState
CurrentStateBlink
CurrentStateColour
CurrentStateDesc
CurrentTime
CurrentTimeFuture
CurrentTimeInvalid
CurrentValue
.BI_1.CurrentValue
Current value of the point.
Relative
OK
Cancel
Expression
79%
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
FlashMode
GradAlpha
GradAngle
New Lay
Edit...
Flowchart...
Clear
Refresh
Close


ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
Expression Flowchart *
TRUE
True
".BI_1.Foreground"
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
FlashMode
GradAlpha
GradAngle
Expression
Relative
".BI_1.CurrentValue"
OK
Cancel
New La
Edit...
Flowchart...
Clear
Refresh
Close

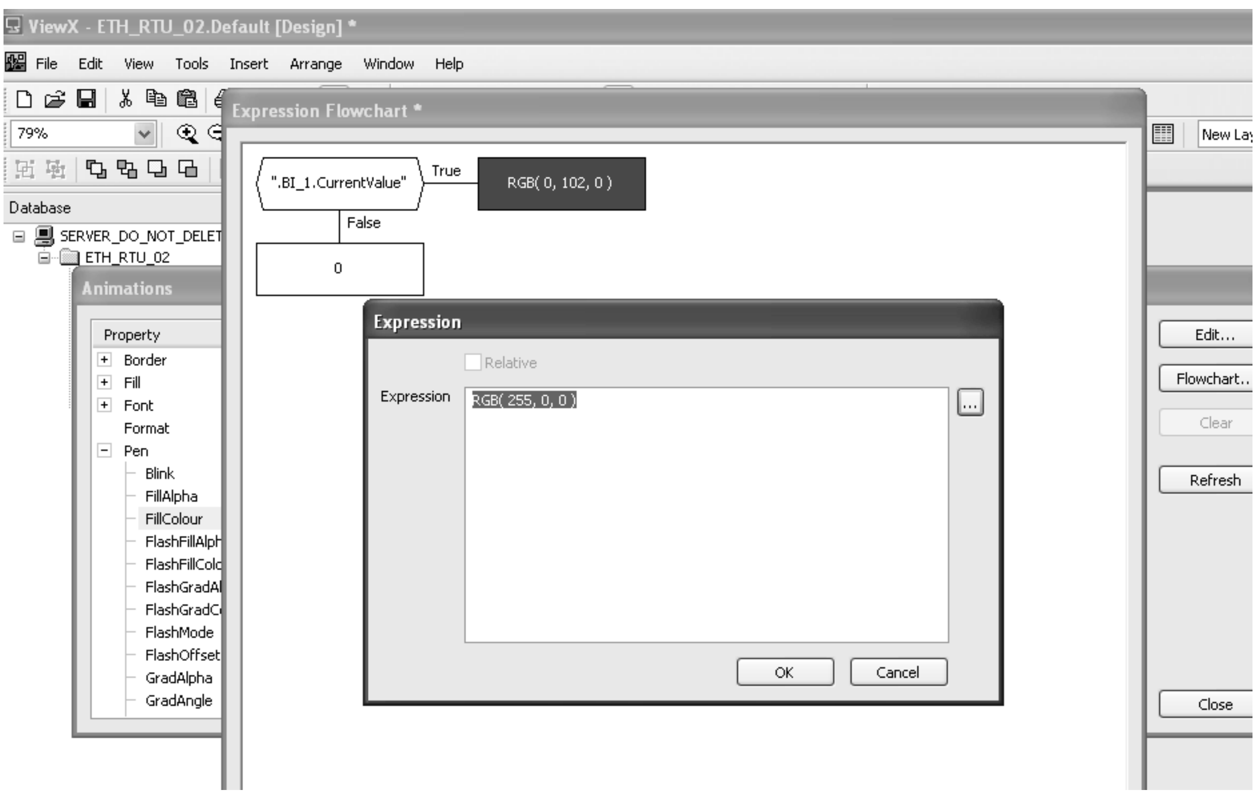

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
False
0
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Edit...
Flowchart...
Clear
Refresh
Close

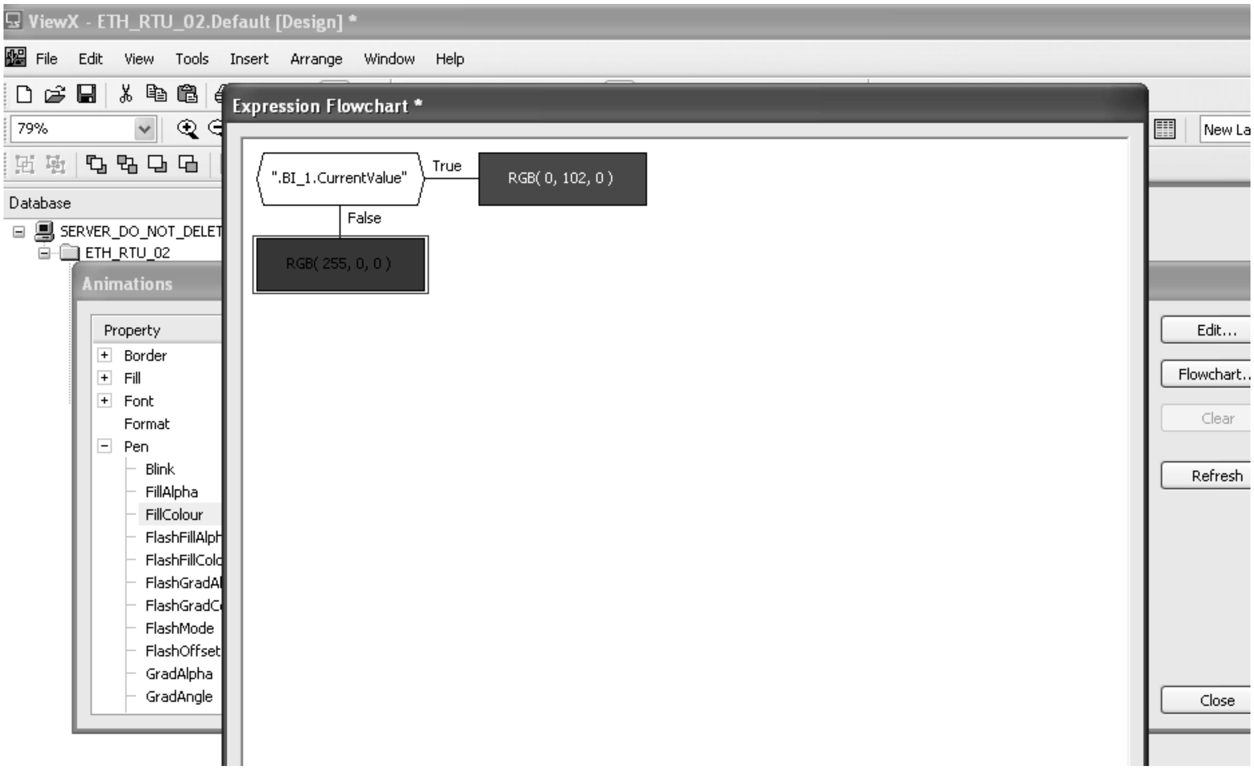

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
0
False
0
Expression
Relative
Expression
OPC Tag...
OPC Historic Tag...
Color
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Edit...
Clear
Refresh
Close

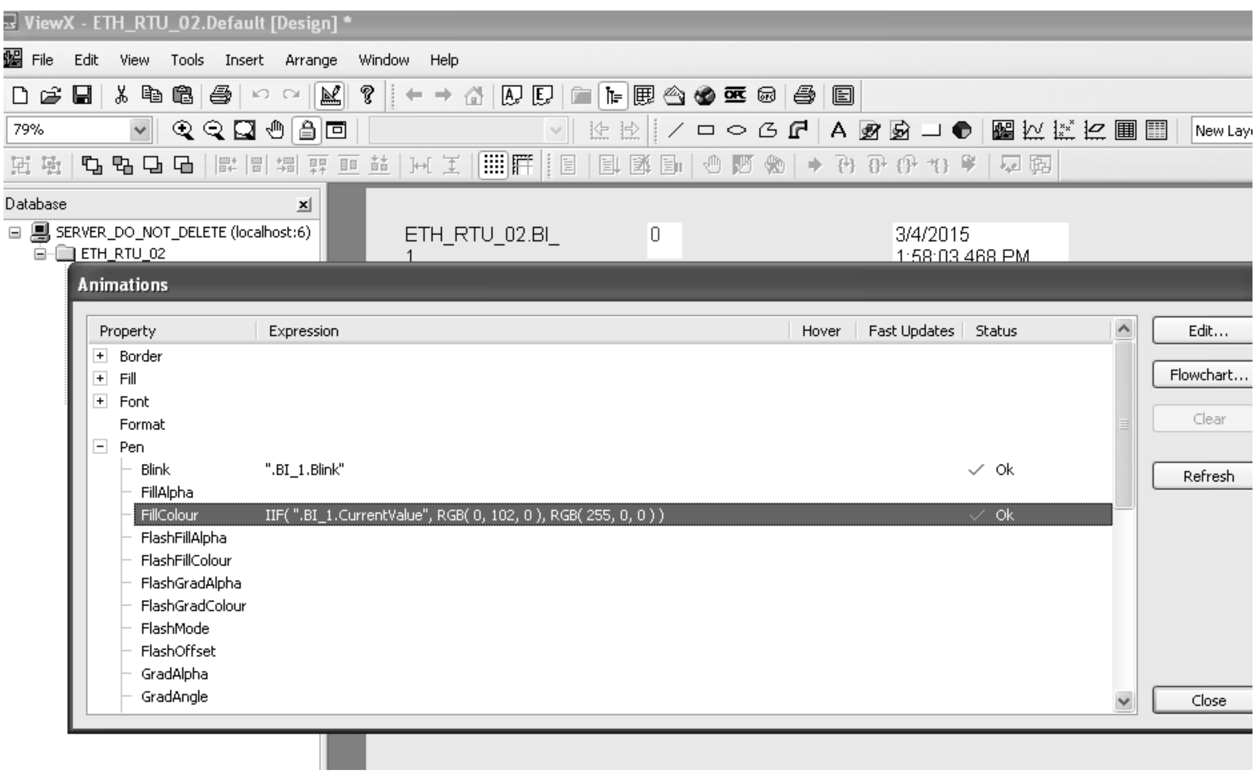

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
0
False
0
Expression
Relative
Expression
RGB(0, 102, 0)
OK
Cancel
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Edit...
Flowchart...
Clear
Refresh
Close

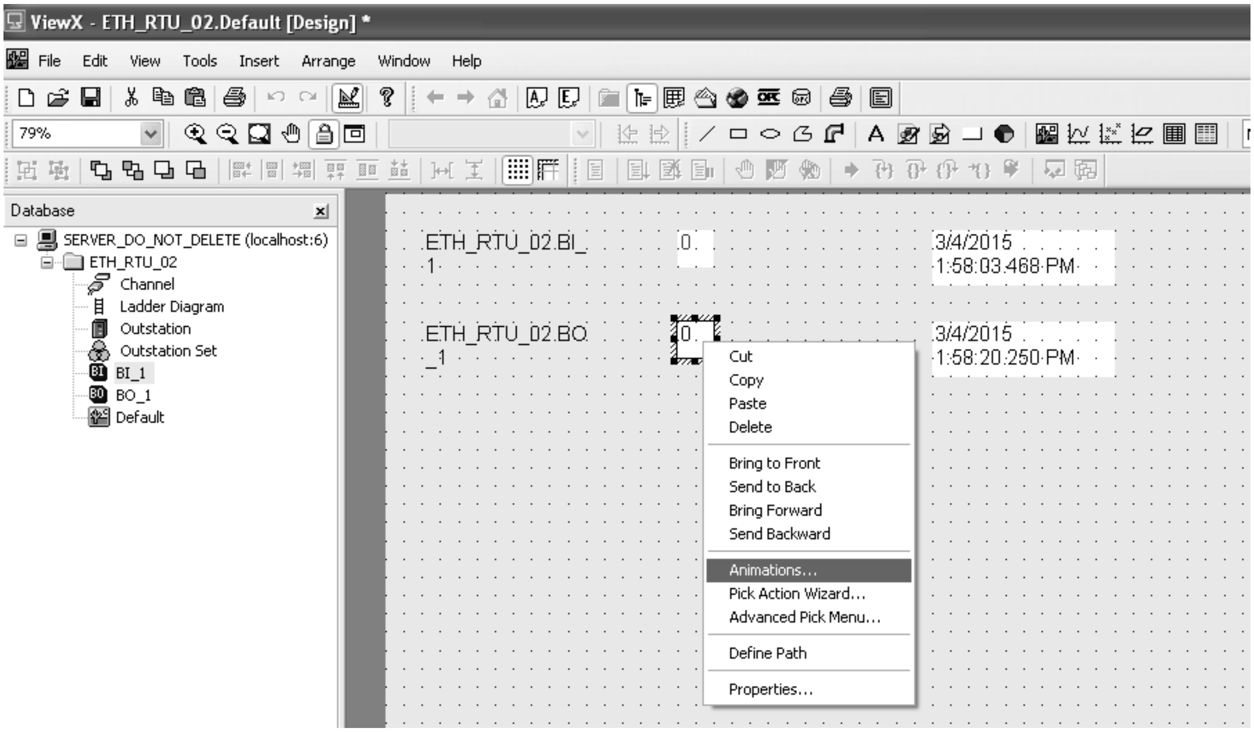

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
RGB(0, 102, 0)
False
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Edit...
Flowchart...
Clear
Refresh
Close

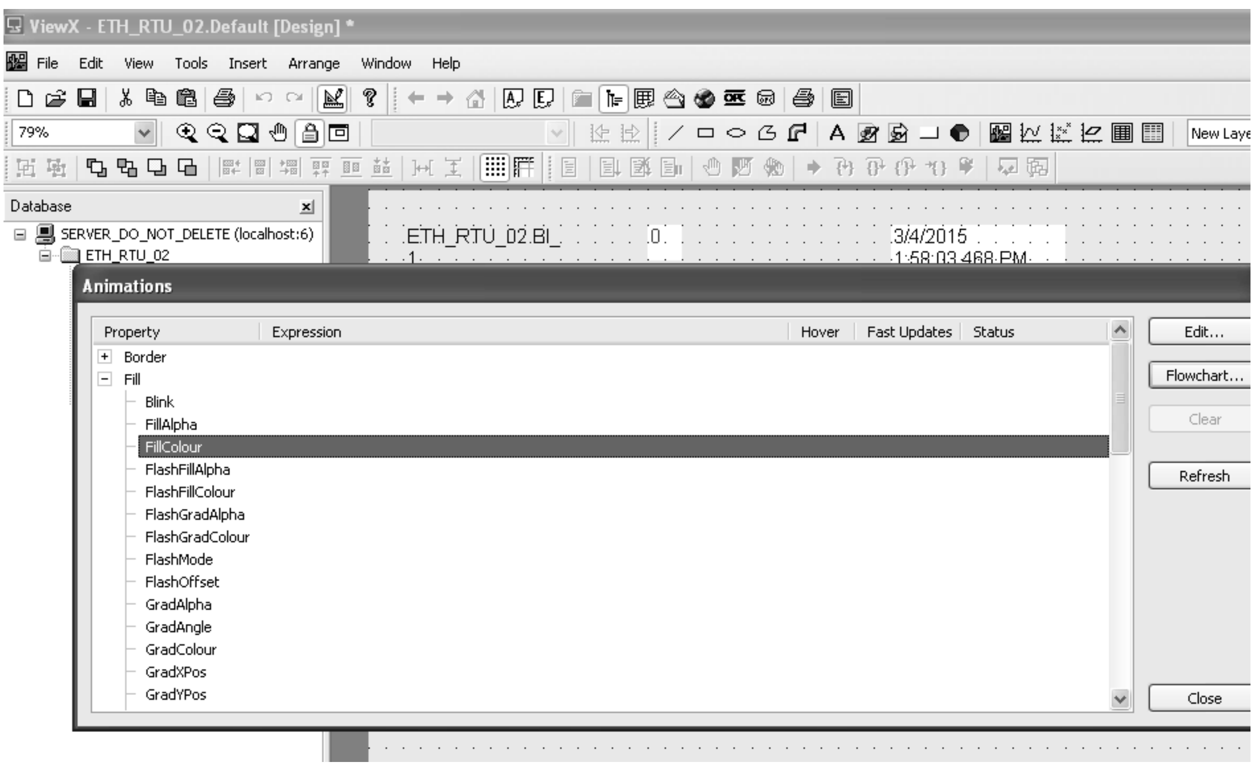

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
RGB(0, 102, 0)
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Expression
Relative
Expression
OPC Tag...
OPC Historic Tag...
Color
Edit...
Flowchart...
Clear
Refresh
Close

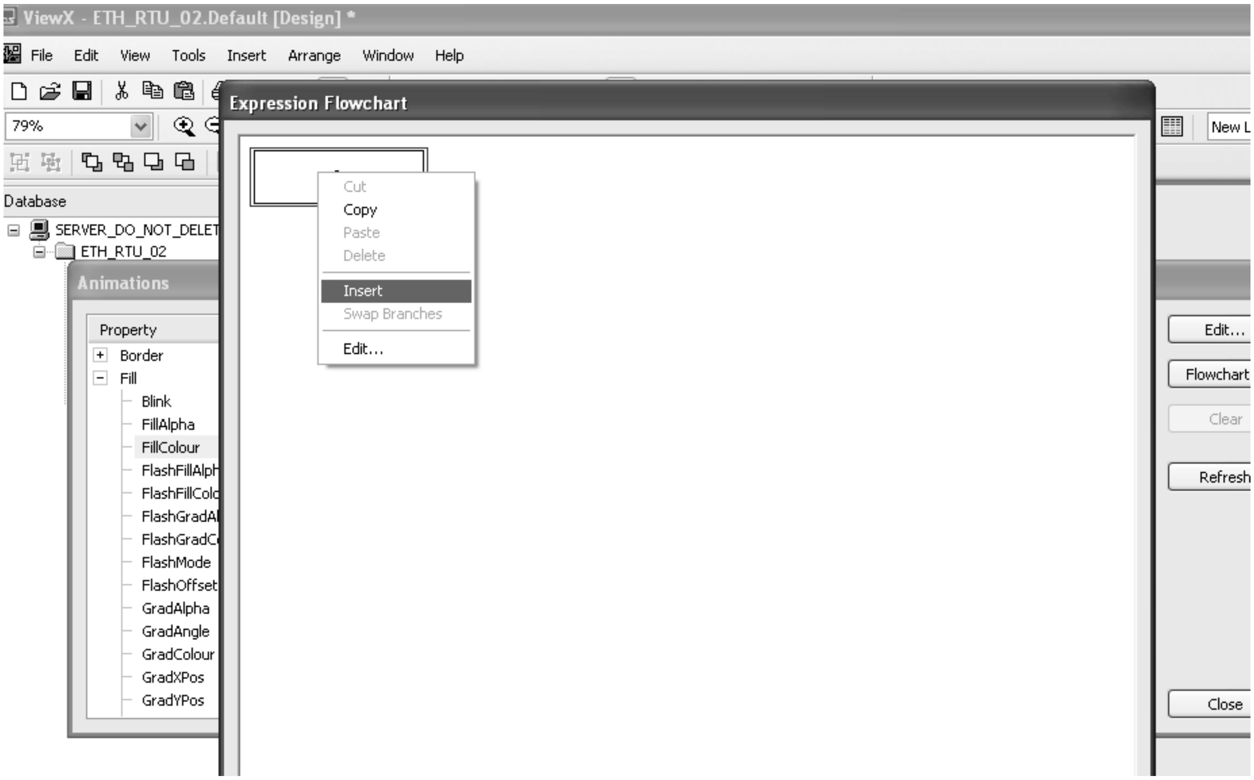

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
RGB(0, 102, 0)
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Expression
Relative
Expression
RGB(255, 0, 0)
OK
Cancel
Edit...
Flowchart...
Clear
Refresh
Close

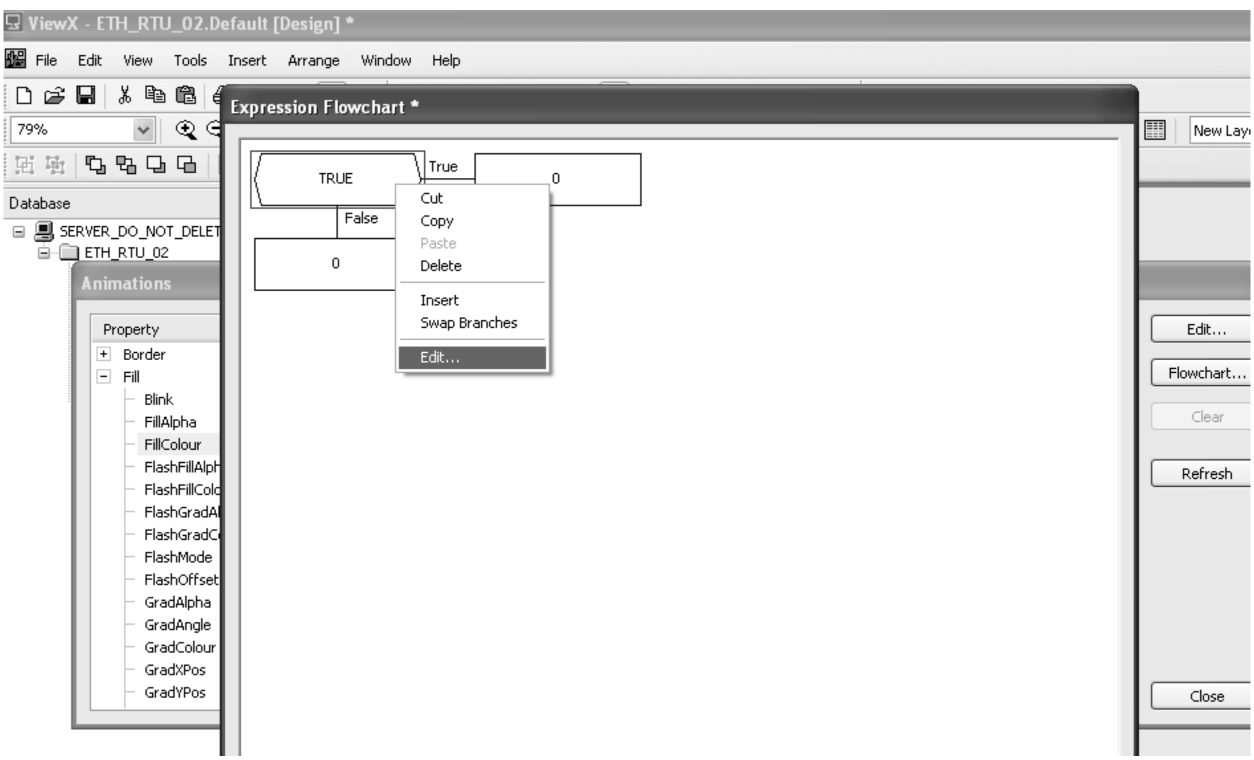

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BI_1.CurrentValue"
True
RGB(0, 102, 0)
False
RGB(255, 0, 0)
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Font
Format
Pen
Blink
FillAlpha
FillColour
Edit...
Flowchart...
Clear
Refresh
Close

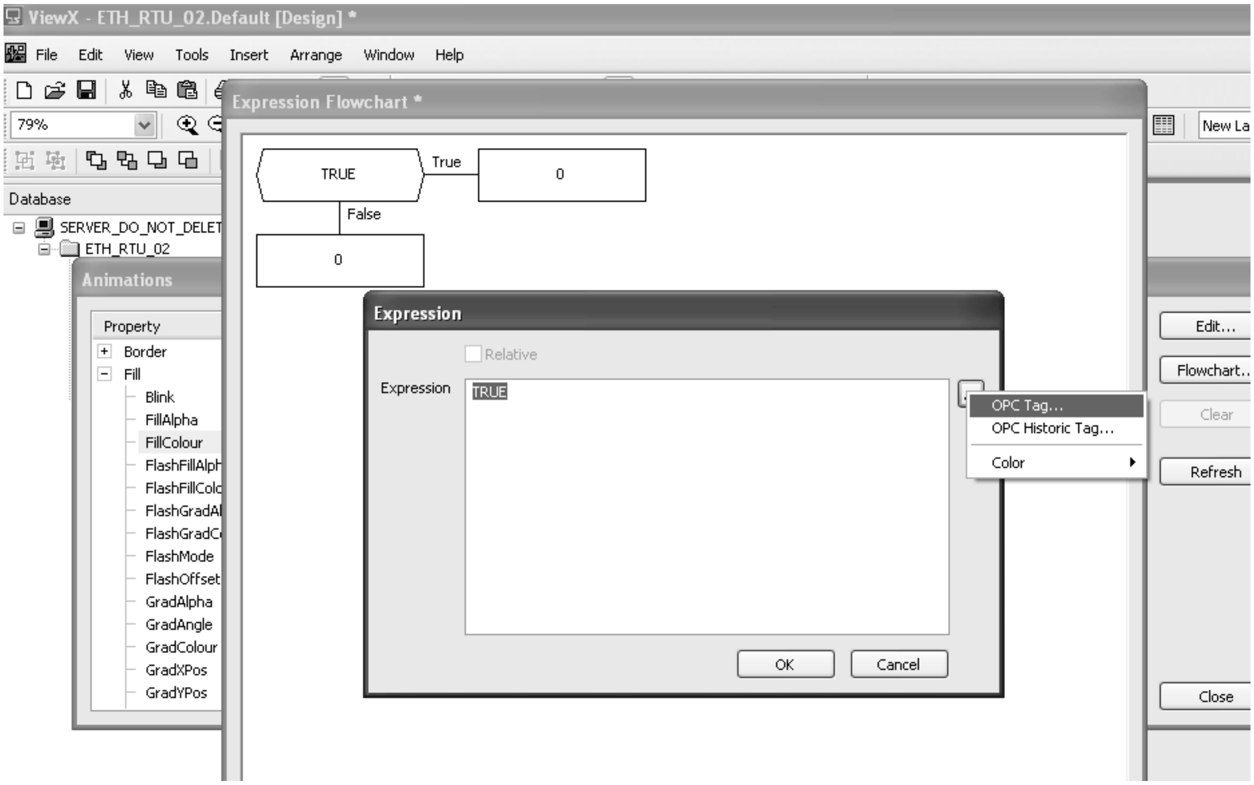

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
ETH_RTU_02.BI_1
0
3/4/2015
Animations
Property Expression Hover Fast Updates Status
Border
Fill
Font
Format
Pen
Blink ".BI_1.Blink" Ok
FillAlpha
FillColour IIF(".BI_1.CurrentValue", RGB(0, 102, 0), RGB(255, 0, 0)) Ok
FlashFillAlpha
FlashFillColour
FlashGradAlpha
FlashGradColour
FlashMode
FlashOffset
GradAlpha
GradAngle
Edit...
Flowchart...
Clear
Refresh
Close

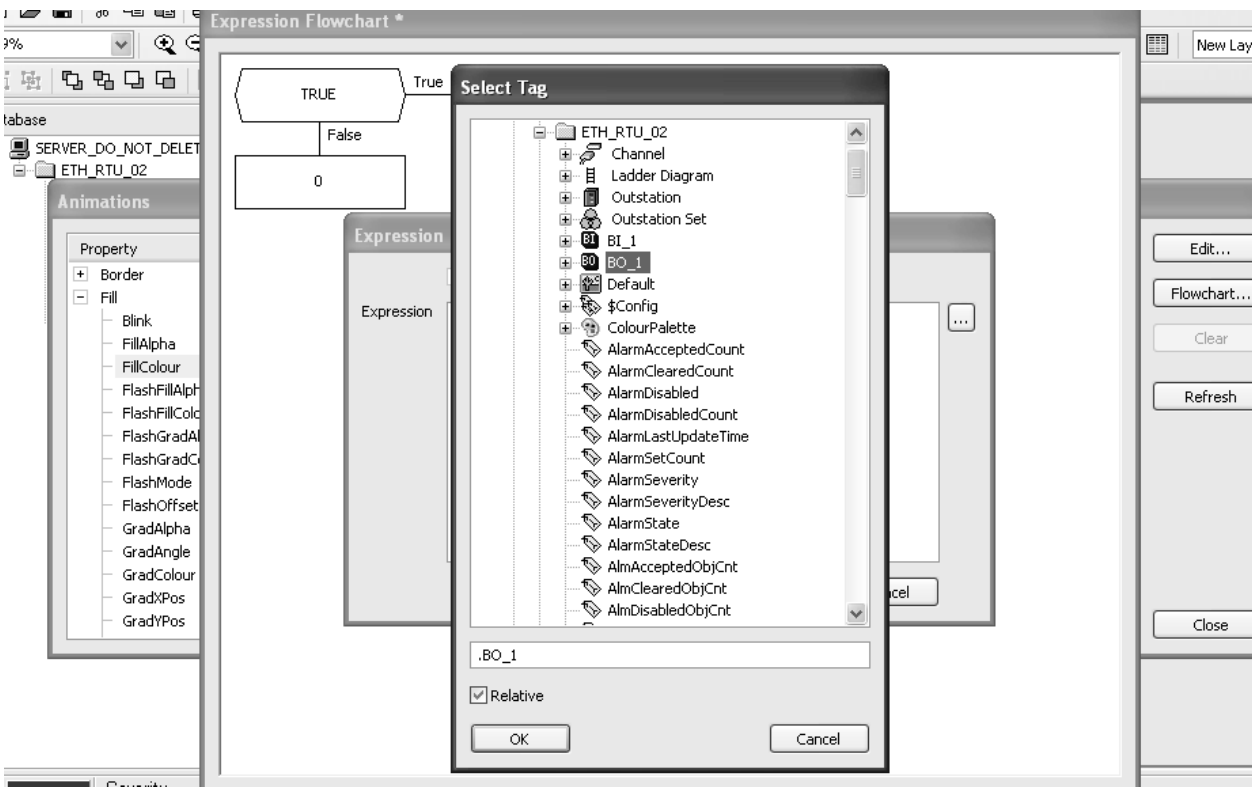

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
ETH_RTU_02.BI_1
0
3/4/2015 1:58:03.468 PM
ETH_RTU_02.BO_1
0
3/4/2015 1:58:20.250 PM
Cut
Copy
Paste
Delete
Bring to Front
Send to Back
Bring Forward
Send Backward
Animations...
Pick Action Wizard...
Advanced Pick Menu...
Define Path
Properties...

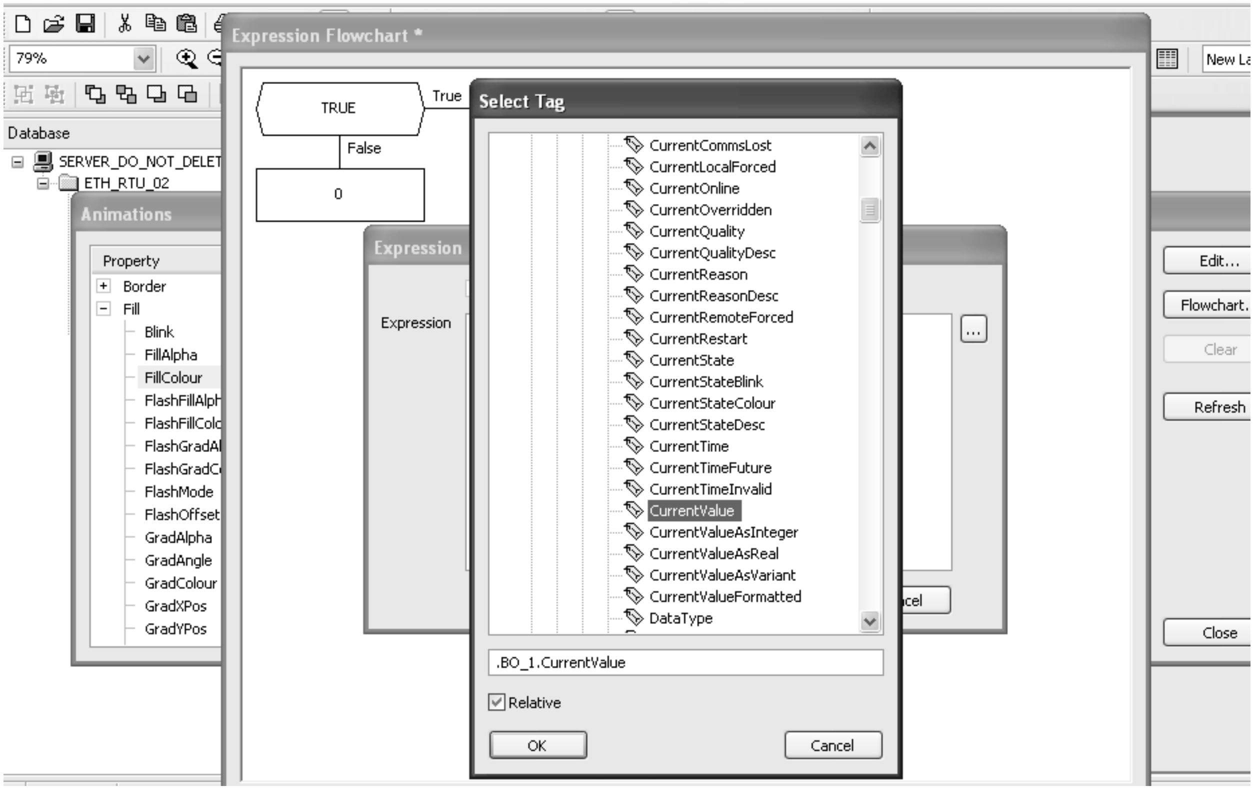

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
New Laye
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
ETH_RTU_02.BI_
0
3/4/2015
Animations
Property
Expression
Hover
Fast Updates
Status
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlpha
FlashFillColour
FlashGradAlpha
FlashGradColour
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

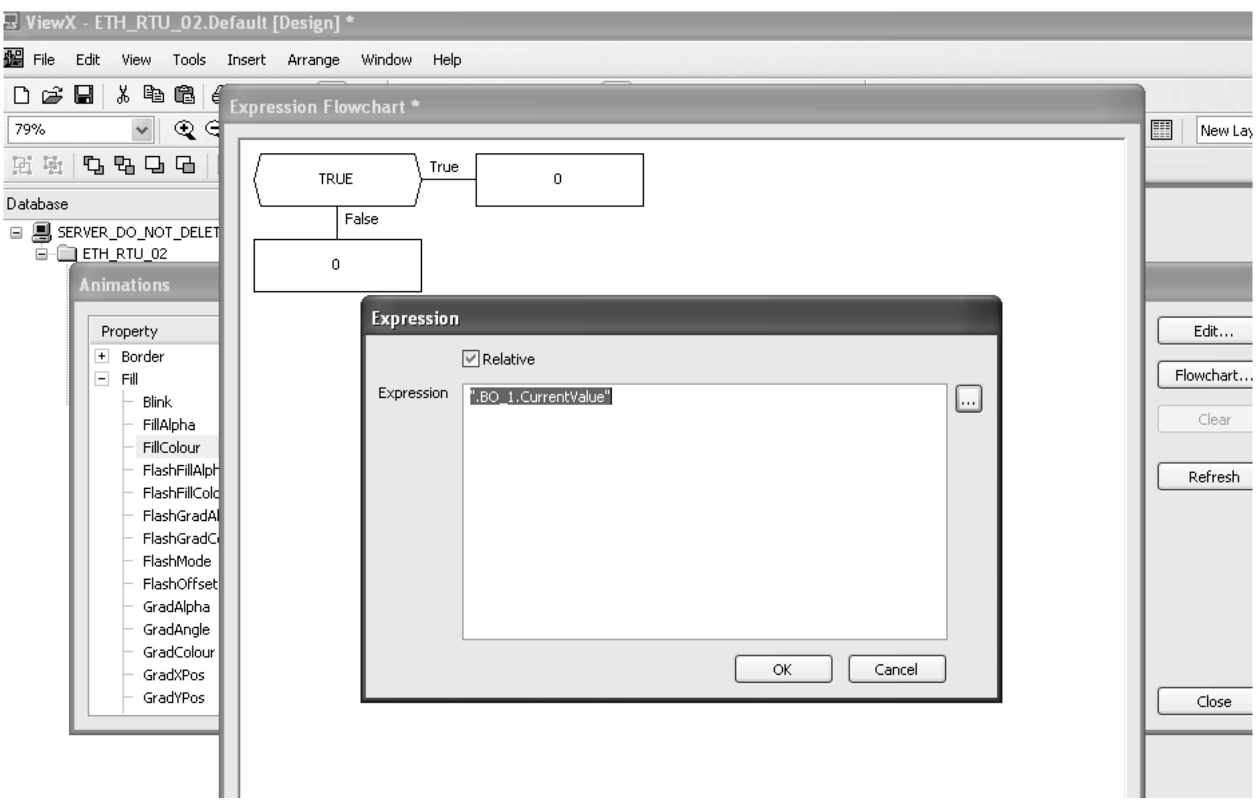

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart
Clear
Refresh
Close

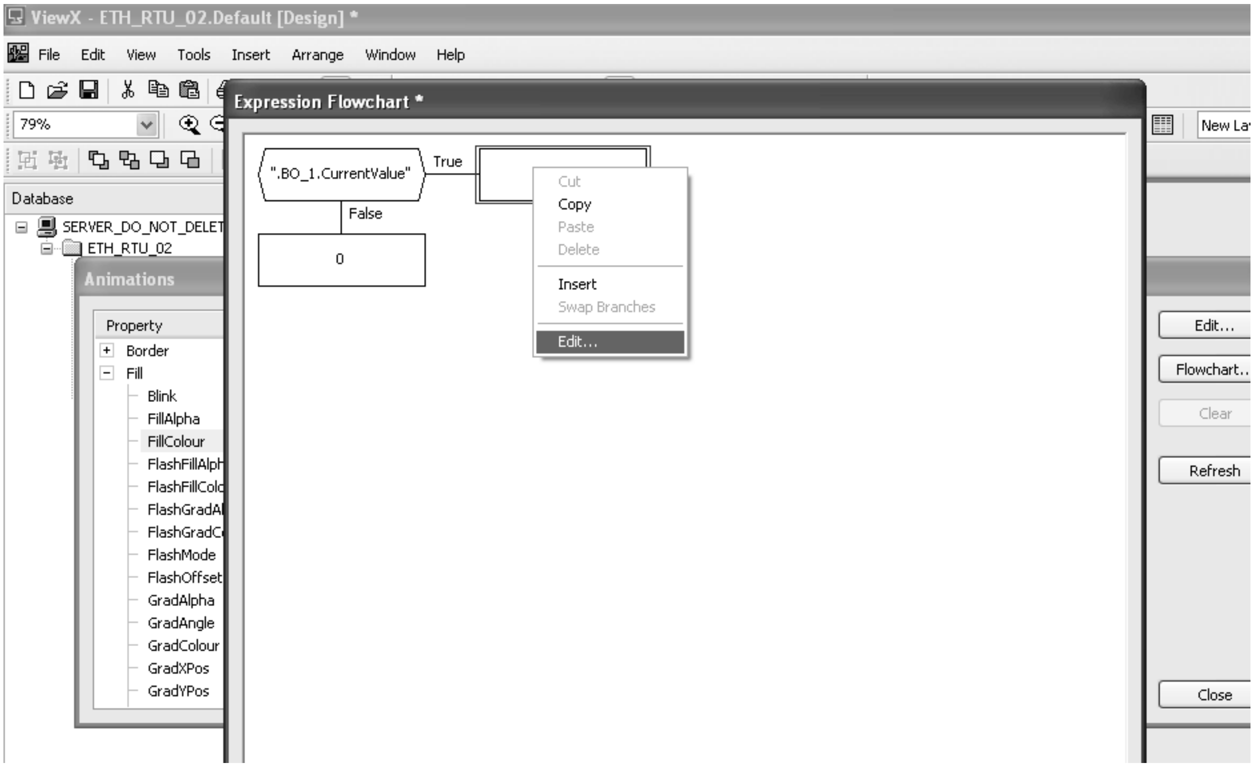

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
TRUE
True
0
False
0
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close


ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlph
FlashFillColo
FlashGradAl
FlashGradC
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Expression Flowchart *
TRUE
True
0
False
0
Expression
Relative
Expression
TRUE
OPC Tag...
OPC Historic Tag...
Color
OK
Cancel
New La
Edit...
Flowchart...
Clear
Refresh
Close

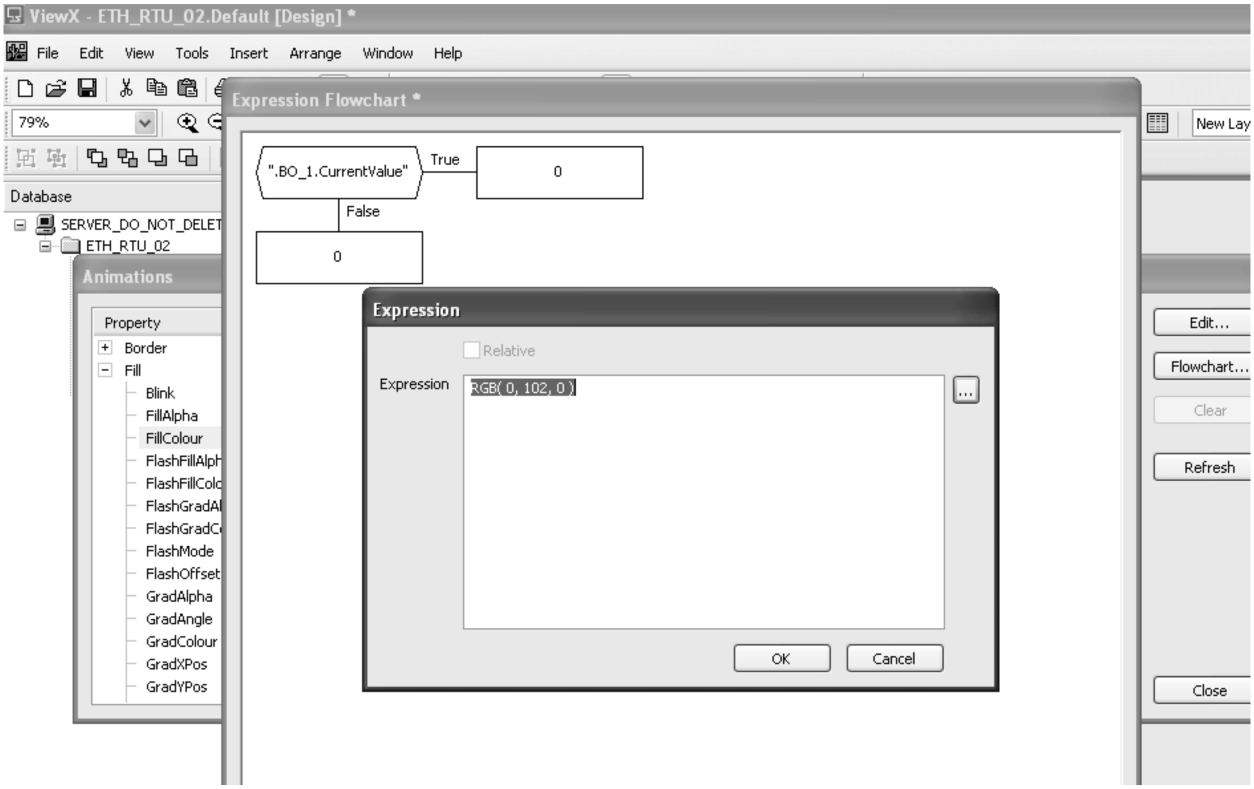

Expression Flowchart *
TRUE
True
False
0
Select Tag
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
$Config
ColourPalette
AlarmAcceptedCount
AlarmClearedCount
AlarmDisabled
AlarmDisabledCount
AlarmLastUpdateTime
AlarmSetCount
AlarmSeverity
AlarmSeverityDesc
AlarmState
AlarmStateDesc
AlmAcceptedObjCnt
AlmClearedObjCnt
AlmDisabledObjCnt
.BO_1
Relative
OK
Cancel
Expression
Expression
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlph
FlashFillColo
FlashGradAl
FlashGradC
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
New Lay
Edit...
Flowchart...
Clear
Refresh
Close

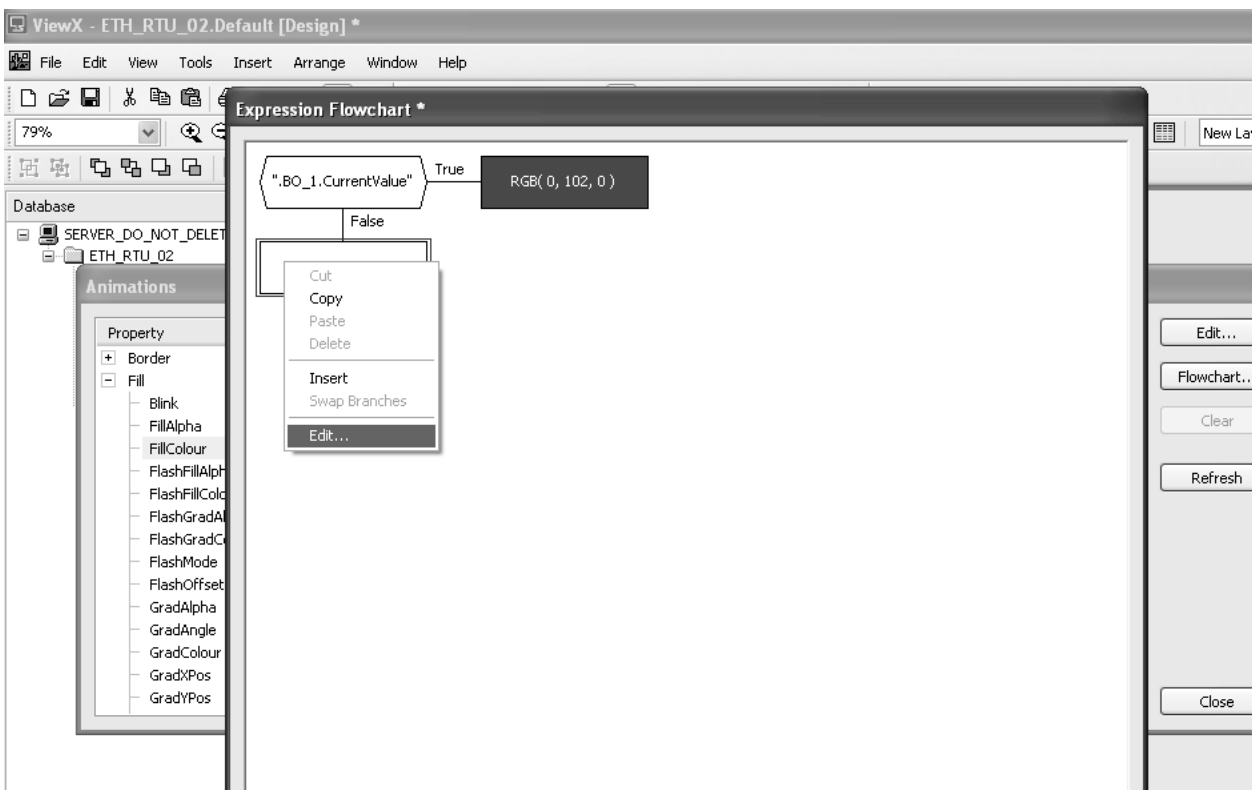

Expression Flowchart *
TRUE
True
False
0
Select Tag
CurrentCommsLost
CurrentLocalForced
CurrentOnline
CurrentOverridden
CurrentQuality
CurrentQualityDesc
CurrentReason
CurrentReasonDesc
CurrentRemoteForced
CurrentRestart
CurrentState
CurrentStateBlink
CurrentStateColour
CurrentStateDesc
CurrentTime
CurrentTimeFuture
CurrentTimeInvalid
CurrentValue
CurrentValueAsInteger
CurrentValueAsReal
CurrentValueAsVariant
CurrentValueFormatted
DataType
.BO_1.CurrentValue
Relative
OK
Cancel
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Refresh
Close


ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
TRUE
True
0
False
0
Expression
Relative
Expression
".BO_1.CurrentValue"
OK
Cancel
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
FlashMode
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

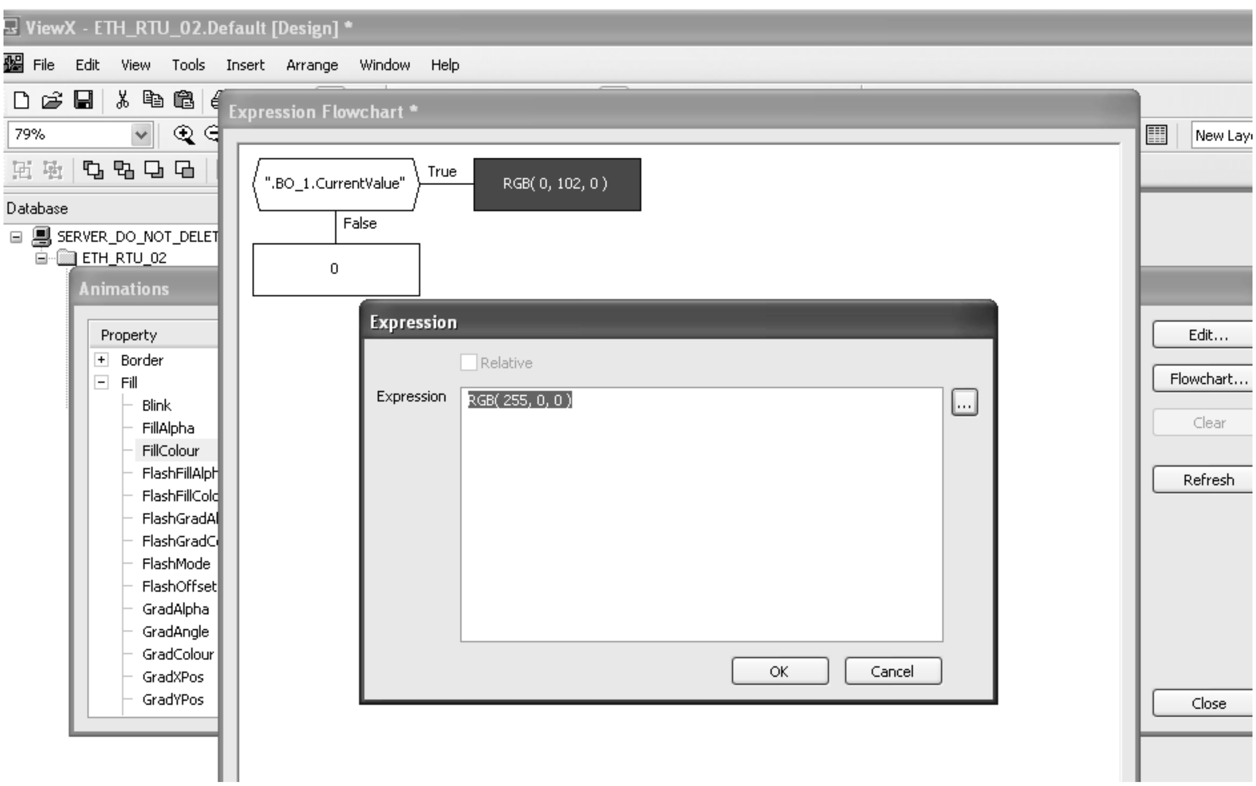

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
".BO_1.CurrentValue"
True
False
0
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

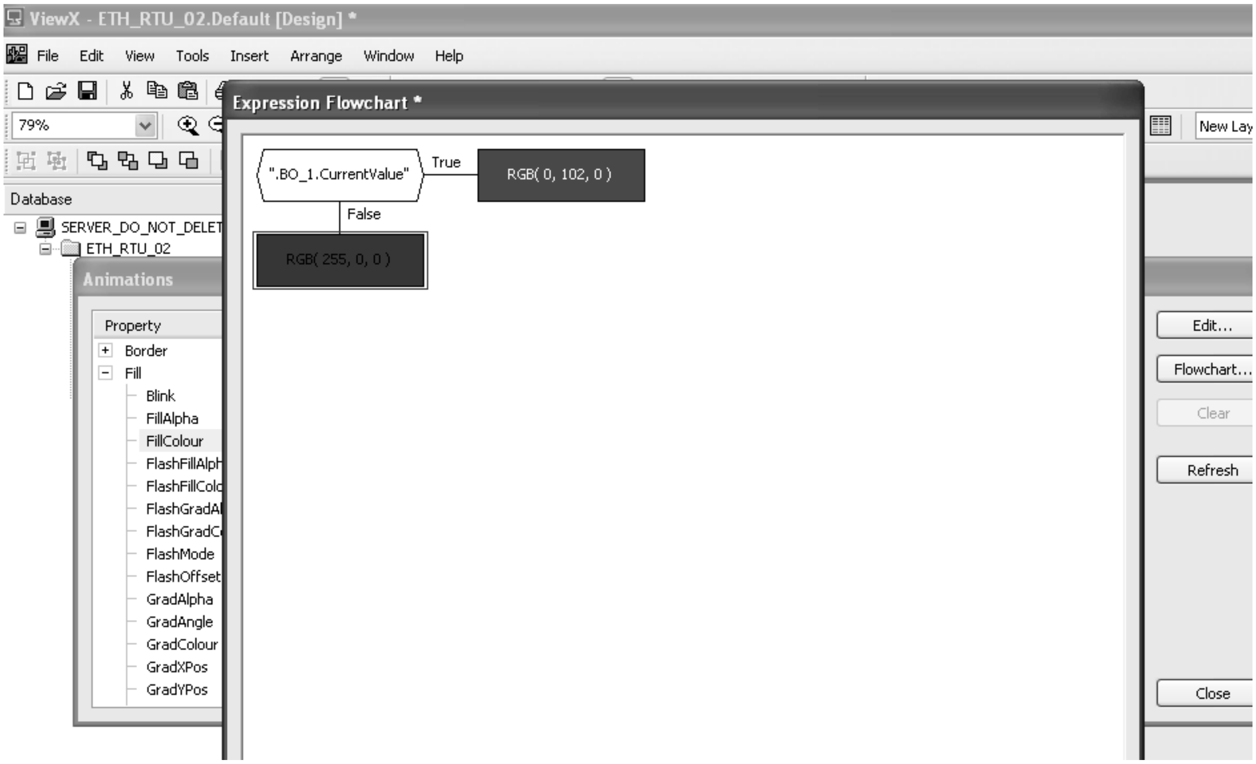

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
".BO_1.CurrentValue"
True
0
False
0
Expression
Relative
Expression
OPC Tag...
OPC Historic Tag...
Color
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart.
Clear
Refresh
Close

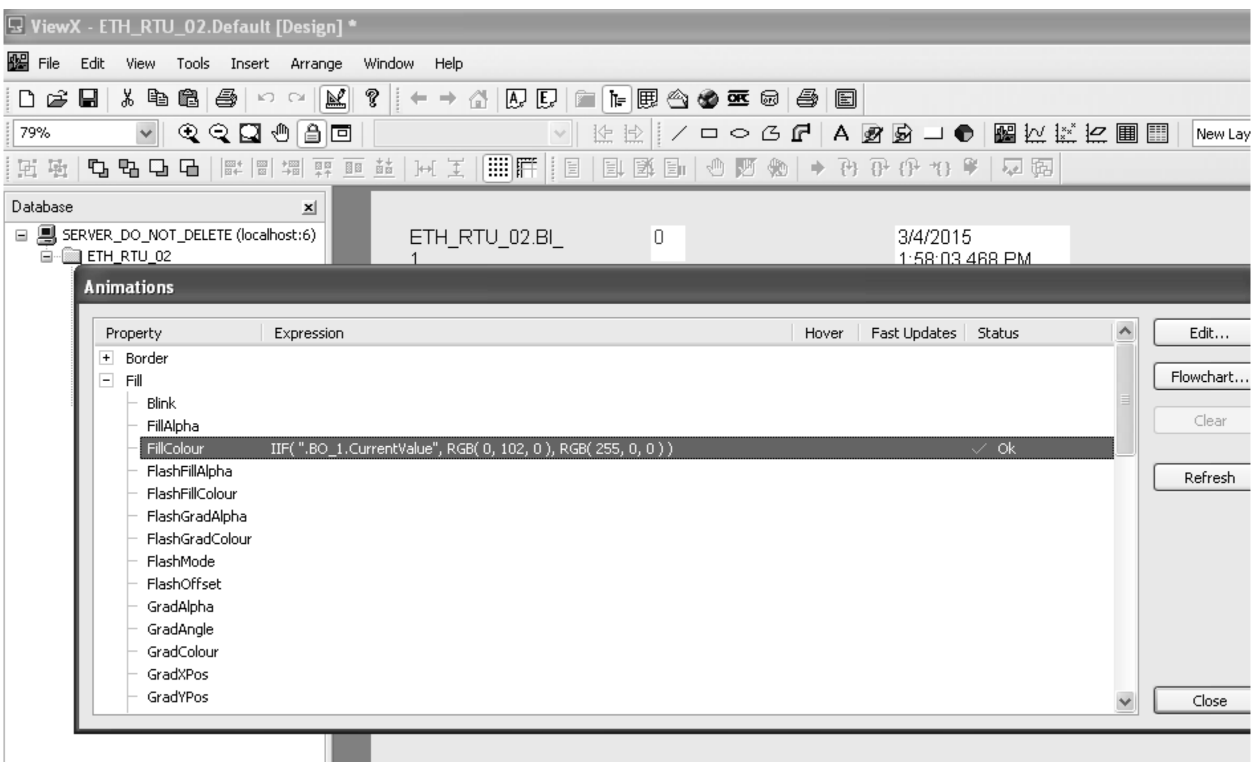

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BO_1.CurrentValue"
True
0
False
0
Expression
Relative
Expression
RGB(0, 102, 0)
OK
Cancel
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

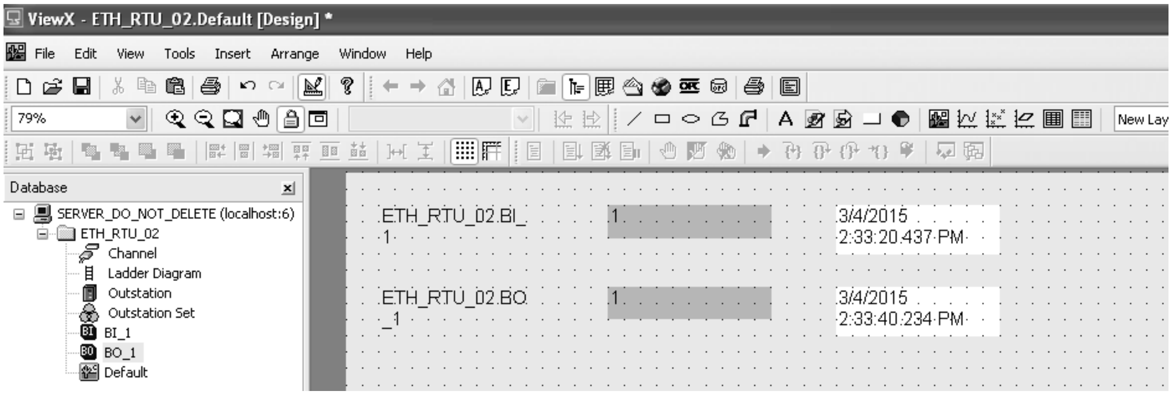

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BO_1.CurrentValue"
True
RGB(0, 102, 0)
False
Cut
Copy
Paste
Delete
Insert
Swap Branches
Edit...
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

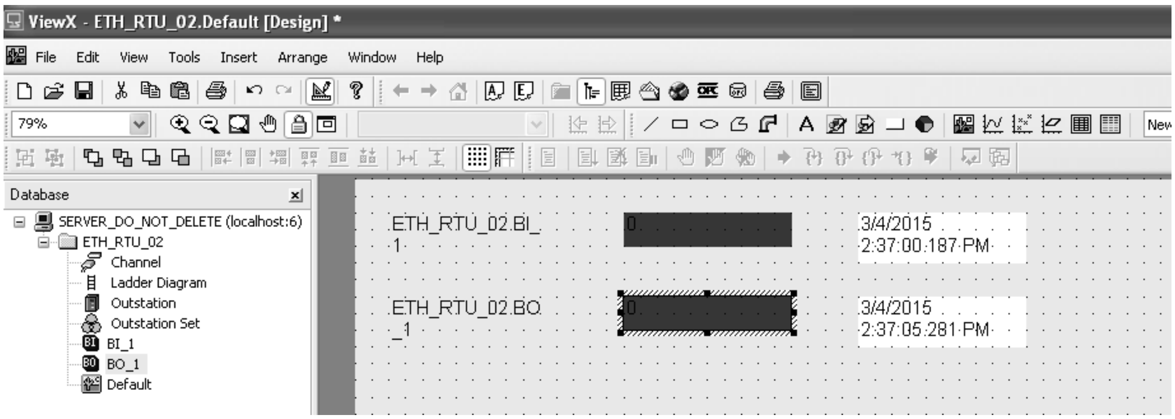

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
".BO_1.CurrentValue"
True
RGB(0, 102, 0)
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlph
FlashFillColo
FlashGradAl
FlashGradC
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Expression
Relative
Expression
0
OPC Tag...
OPC Historic Tag...
Color
New Lay
Edit...
Flowchart...
Clear
Refresh
Close

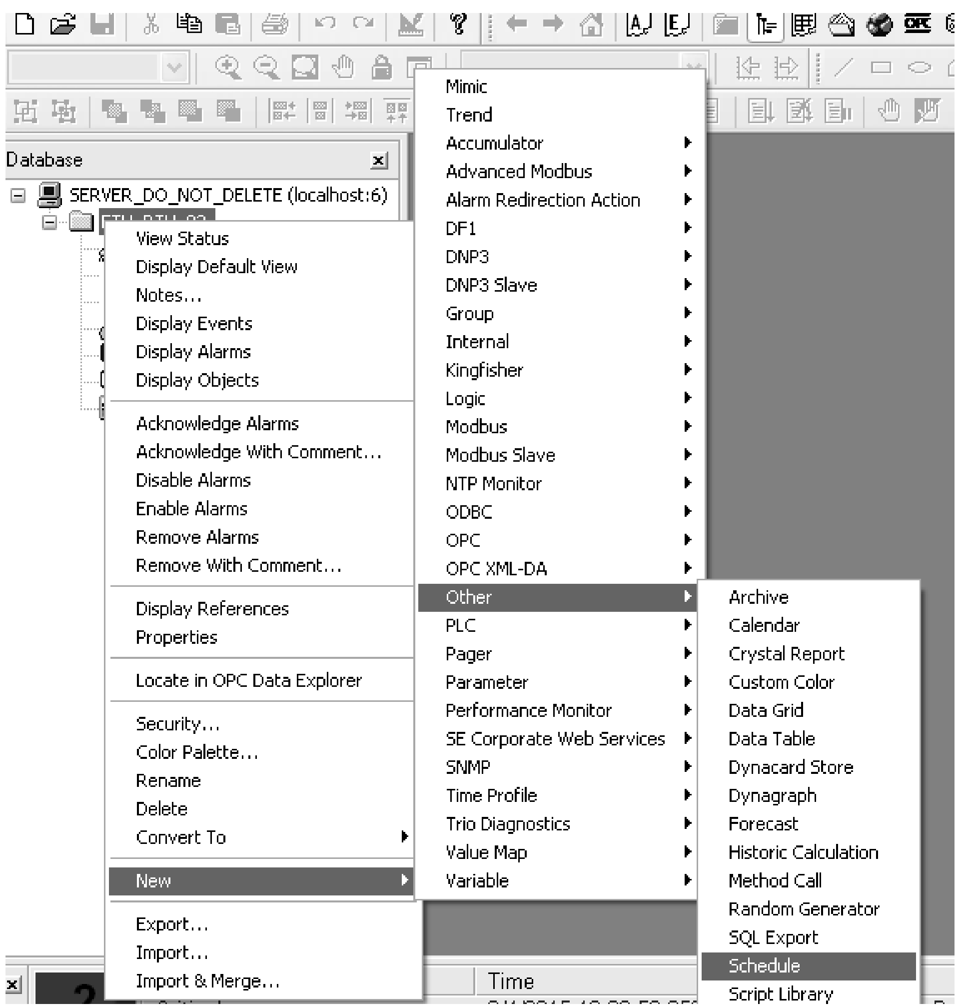

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
79%
Expression Flowchart *
".BO_1.CurrentValue"
True
RGB(0, 102, 0)
False
0
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
FlashFillAlph
FlashFillColo
FlashGradAl
FlashGradC
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Expression
Relative
Expression
RGB(255, 0, 0)
OK
Cancel
New Laye
Edit...
Flowchart...
Clear
Refresh
Close

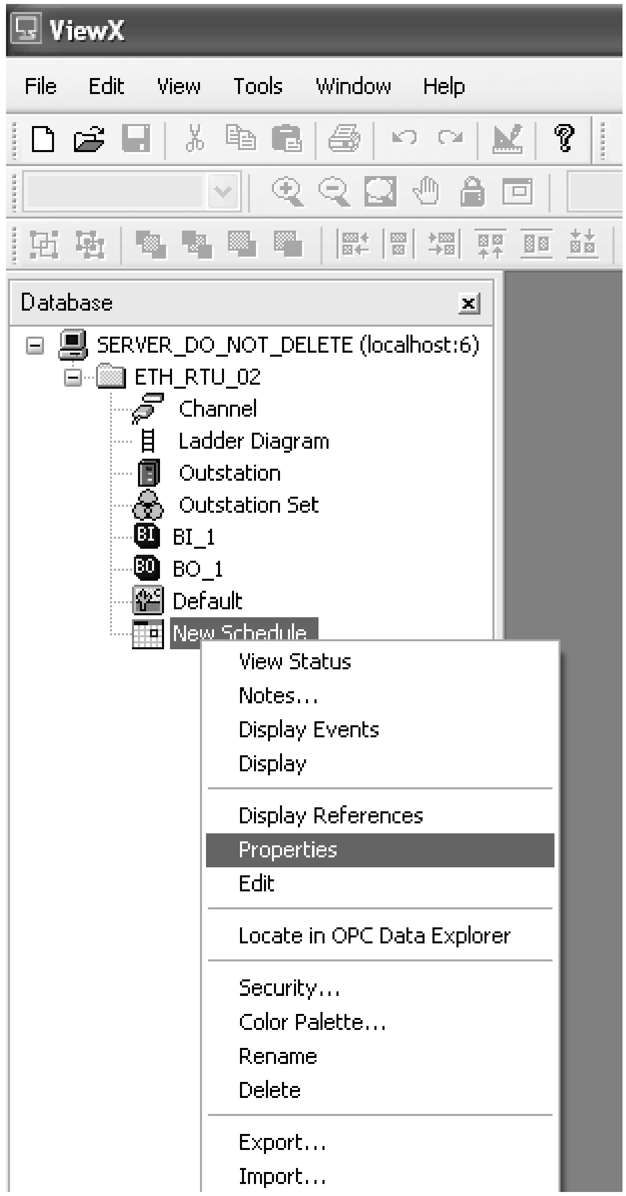

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Expression Flowchart *
".BO_1.CurrentValue"
True
RGB(0, 102, 0)
False
RGB(255, 0, 0)
Database
SERVER_DO_NOT_DELET
ETH_RTU_02
Animations
Property
Border
Fill
Blink
FillAlpha
FillColour
Edit...
Flowchart...
Clear
Refresh
Close

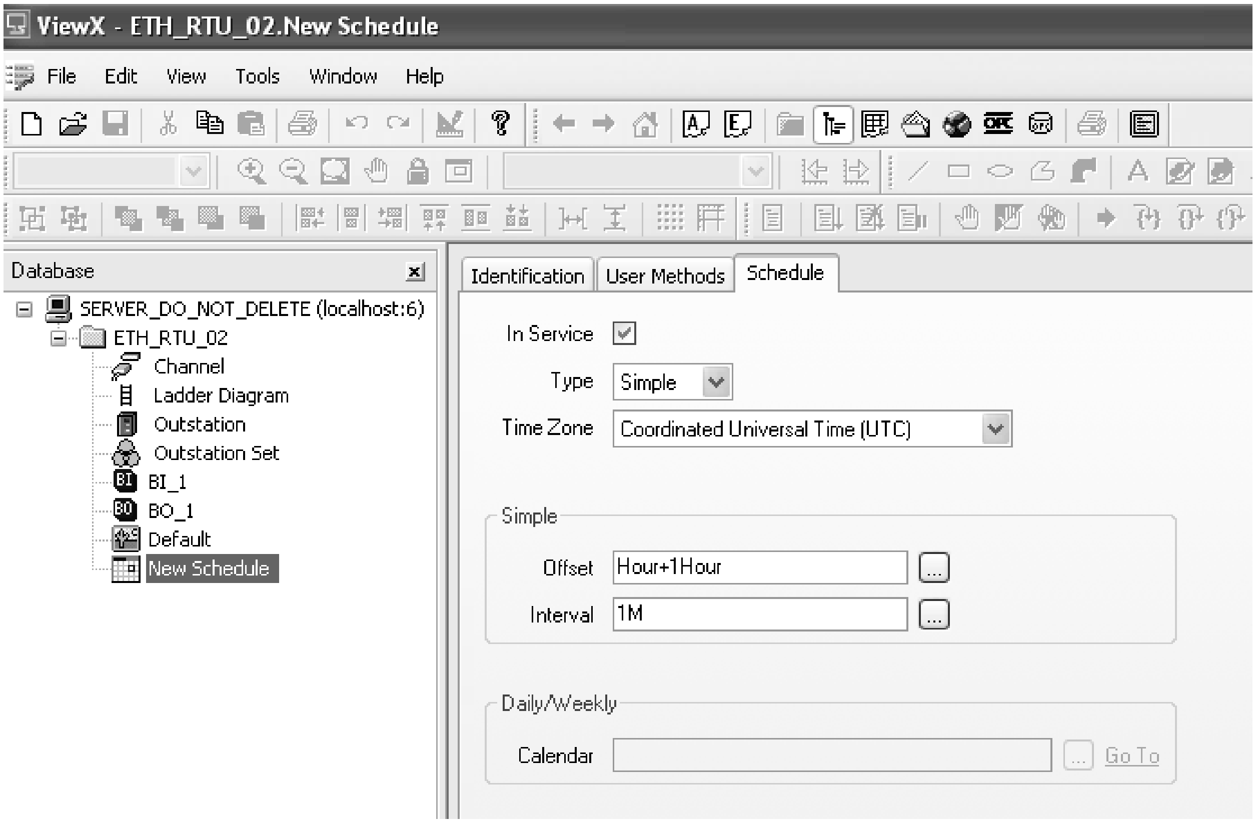

ViewX - ETH_RTU_02.Default [Design] *
File Edit View Tools Insert Arrange Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
ETH_RTU_02.BI_1
0
3/4/2015 1:58:03.468 PM
Animations
Property
Expression
Hover
Fast Updates
Status
Border
Fill
Blink
FillAlpha
FillColour
IIF(".BO_1.CurrentValue", RGB(0, 102, 0), RGB(255, 0, 0))
Ok
FlashFillAlpha
FlashFillColour
FlashGradAlpha
FlashGradColour
FlashMode
FlashOffset
GradAlpha
GradAngle
GradColour
GradXPos
GradYPos
Edit...
Flowchart...
Clear
Refresh
Close

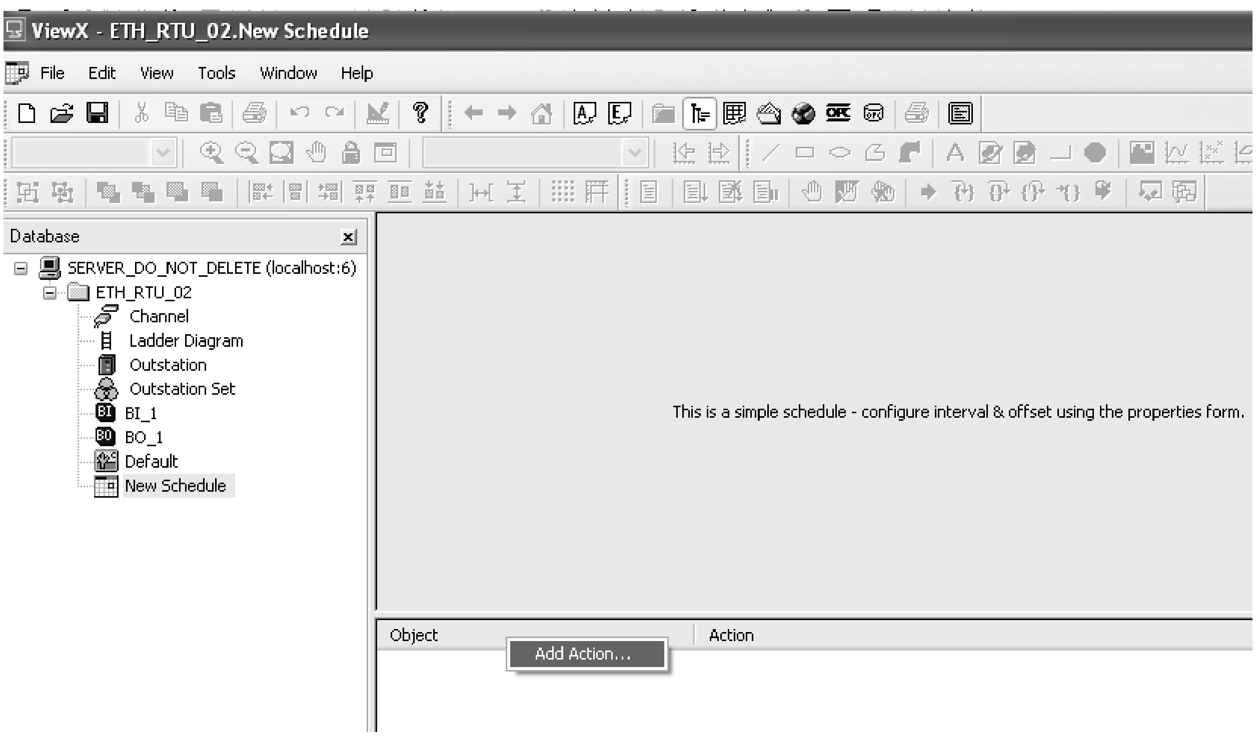

ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
New Lay
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
ETH_RTU_02.BI_1
1
3/4/2015 2:33:20.437 PM
ETH_RTU_02.BO_1
1
3/4/2015 2:33:40.234 PM

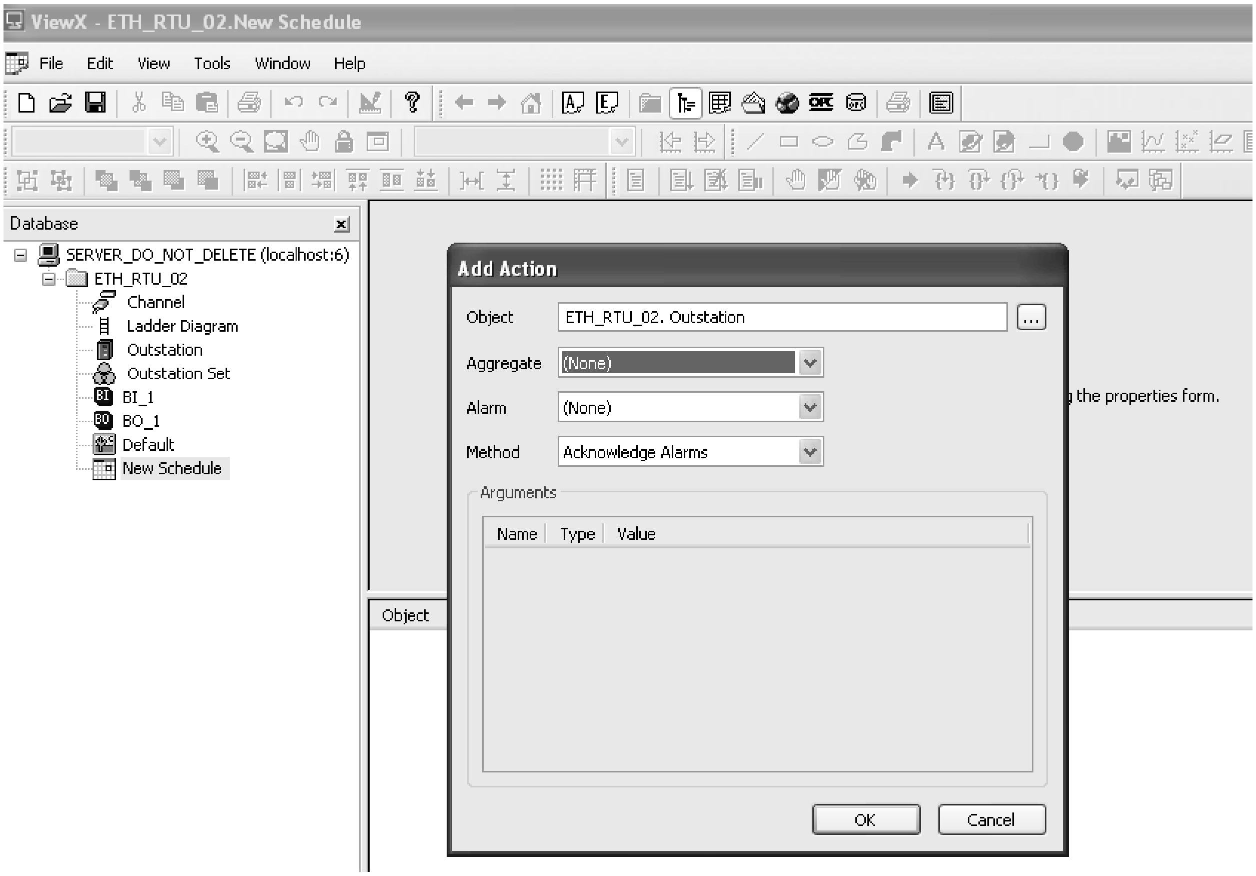

ViewX - ETH_RTU_02.Default [Design] *
File
Edit
View
Tools
Insert
Arrange
Window
Help
79%
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
ETH_RTU_02.BI_1
0
3/4/2015 2:37:00.187 PM
ETH_RTU_02.BO_1
0
3/4/2015 2:37:05.281 PM

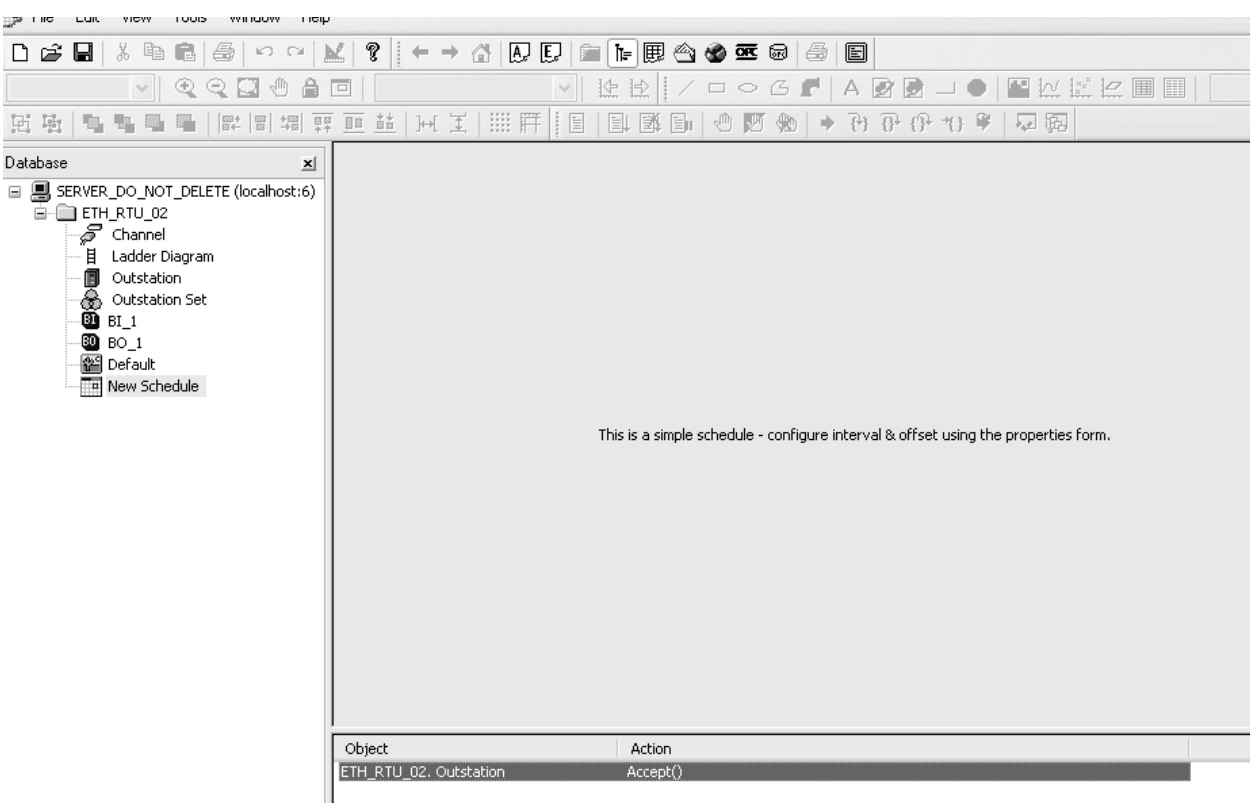

Database
SERVER_DO_NOT_DELETE (localhost:6)
View Status
Display Default View
Notes...
Display Events
Display Alarms
Display Objects
Acknowledge Alarms
Acknowledge With Comment...
Disable Alarms
Enable Alarms
Remove Alarms
Remove With Comment...
Display References
Properties
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Convert To
New
Export...
Import...
Import & Merge...
Mimic
Trend
Accumulator
Advanced Modbus
Alarm Redirection Action
DF1
DNP3
DNP3 Slave
Group
Internal
Kingfisher
Logic
Modbus
Modbus Slave
NTP Monitor
ODBC
OPC
OPC XML-DA
Other
PLC
Pager
Parameter
Performance Monitor
SE Corporate Web Services
SNMP
Time Profile
Trio Diagnostics
Value Map
Variable
Archive
Calendar
Crystal Report
Custom Color
Data Grid
Data Table
Dynacard Store
Dynagraph
Forecast
Historic Calculation
Method Call
Random Generator
SQL Export
Schedule
Script Library
Time


ViewX
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
New Schedule
View Status
Notes...
Display Events
Display
Display References
Properties
Edit
Locate in OPC Data Explorer
Security...
Color Palette...
Rename
Delete
Export...
Import...


ViewX - ETH_RTU_02.New Schedule
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
New Schedule
Identification
User Methods
Schedule
In Service
Type
Simple
Time Zone
Coordinated Universal Time (UTC)
Simple
Offset
Hour+1Hour
Interval
1M
Daily/Weekly
Calendar
Go To


ViewX - ETH_RTU_02.New Schedule
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
New Schedule
This is a simple schedule - configure interval & offset using the properties form.
Object
Action
Add Action...


ViewX - ETH_RTU_02.New Schedule
File Edit View Tools Window Help
Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
New Schedule
Add Action
Object ETH_RTU_02. Outstation
Aggregate (None)
Alarm (None)
Method Acknowledge Alarms
Arguments
Name Type Value
OK Cancel
Object
the properties form.


Database
SERVER_DO_NOT_DELETE (localhost:6)
ETH_RTU_02
Channel
Ladder Diagram
Outstation
Outstation Set
BI_1
BO_1
Default
New Schedule
This is a simple schedule - configure interval & offset using the properties form.
Object Action
ETH_RTU_02. Outstation Accept()