**Vaughn College of Aeronautics and Technology**

**MCE 310L-1 Fundamentals of Mechatronics Laboratory**

**Spring Semester 2019**

**Lab 1**

**Logic Operations in PLC Programs**

**Prepared for:**

**Dr. Miguel Bustamante**

**Experiment conducted by:**

**Jason Becker**

**Omomhene Eimunjeze**

**Tylar Giraud**

**Date of Experiment:**

**2/7/2019**

**Date of Written Report:**

**Table of Contents**

[**Objective**](#_r1582v923x75) **3**

[EQUIPMENT](#_pxxn33g5xu7s) 3

[RESULTS](#_iaws1fqbev1h) 3

[Exercise 1](#_8vpc83mmaixu) 3

[EXERCISE 2](#_uq3b1o1sytmg) 5

[Conclusion](#_hxxxamdhhxim) 10

[References](#_f10tvtfrf5ag) 10

# 

# **Objective**

The objective of this lab was to get familiar with the Instruction List (IL) and Structured Text(ST) coding language in Lab Soft, in addition, we are using the UnTrainer to test and run our said code. We were also supposed to code using AND, OR, XOR, NOT, NOR and NAND logic operators as well as Set and Reset dominance.

## **EQUIPMENT**

* LUCAS-NUELLE Labsoft, PLC and Bus technology
* UniTrain Experimenter SO4203-2B

### **RESULTS**

### Exercise 1

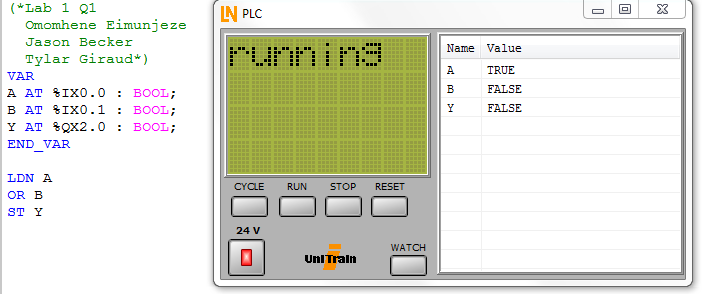


Image 1- Code and Results for Exercise 1

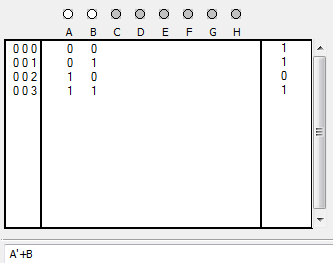
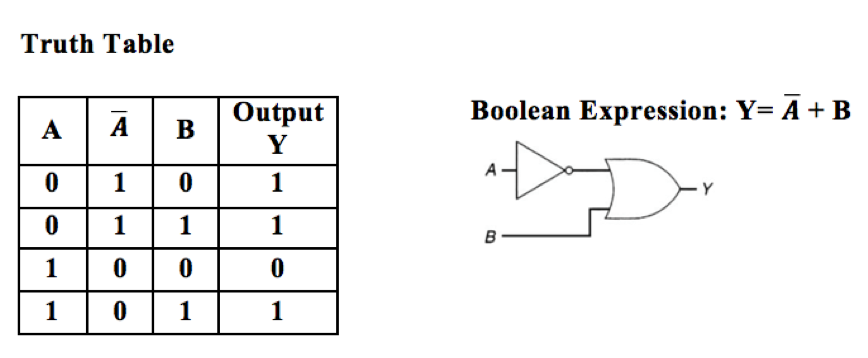
****

Image 3 and 4- Problem 1 and the truth table

Explanation

The code above shows a simple IL code using an inverter for the A output and using test the results using an OR gate. The truth table next to the picture shows all the instants in which the output would be 1. We assigningsymbolic addresses A, and B to the physical address %IX0.0, %IX0.1, which requires an input of 1-bit of memory respectively allocated at 0 bytes 0 bit and 0 bytes 1 bit respectively, beginning from right-side in PLC.

Also, symbolic address Y is assigned to the physical address %QX2.0 which requires an output of 1-bit of memory allocated at 2 bytes 0 bit beginning from right-side in PLC. It is observed that Y is true(1), when and B are both true(1) (A is false), or either =1/(A=0) and B=0 or =0/(A=1) and B=1. On the other hand, the value observed at the physical address Y is false(0) when both and B are false(0), (A=1).

### EXERCISE 2

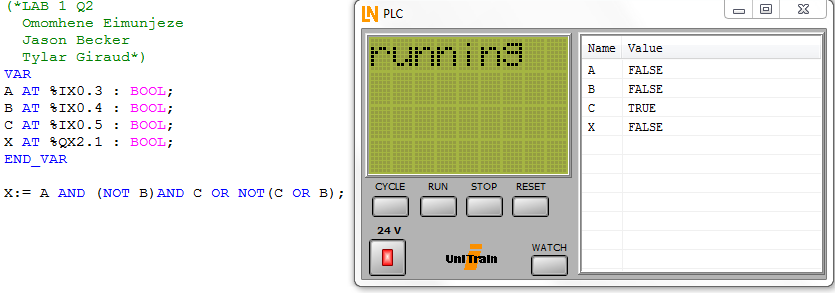


Image 3- Code for Exercise 2

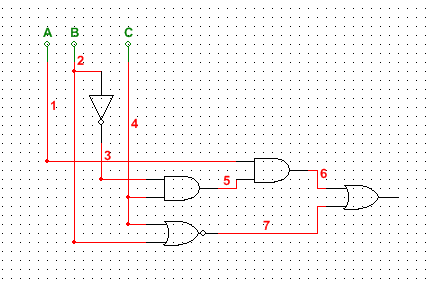
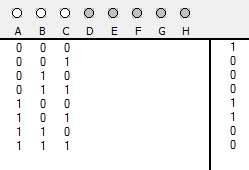


Image 4- Truth Table for problem 2 Image 5- Logic Gates

EXPLANATION

The two images above are the code and truth table for the function given.

X=A\*B’\*C+(C+B)’

Here, we assignedsymbolic addresses A, B, and C to the physical address %IX0.3, %IX0.4, %IX0.5 respectively which requires an input of 1-bit of memory allocated at (0 bytes 3 bit) and (0 bytes 4 bit) and (0 bytes 5 bit) respectively, beginning from right-side in PLC. Also, symbolic address X is assigned to the physical address %QX2.1 which requires an output of 1-bit of memory allocated at (2 bytes 1 bit) beginning from right-side in PLC. The program loads the following logic function:

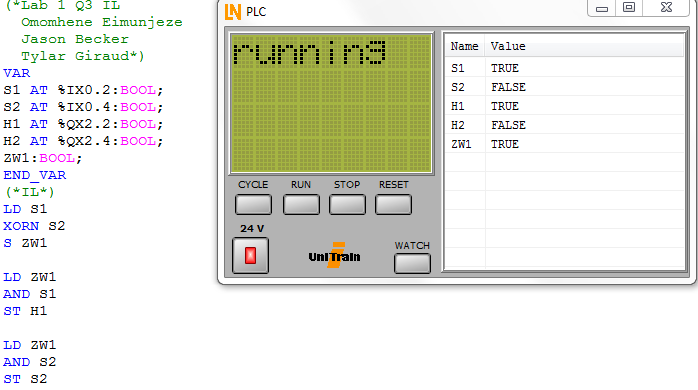
X= A..C +

Output X is False (X=0) when: A=0, =1/(B=0), C=0. Implementing the Boolean expression 0f X, output X will be equal to 0.

**A..C** = (0 AND 1 AND 1) = 0

= = 0

Therefore, output X = 0 + 0 = 0 = False

EXERCISE 3Image 6- Code and Results for Exercise 3

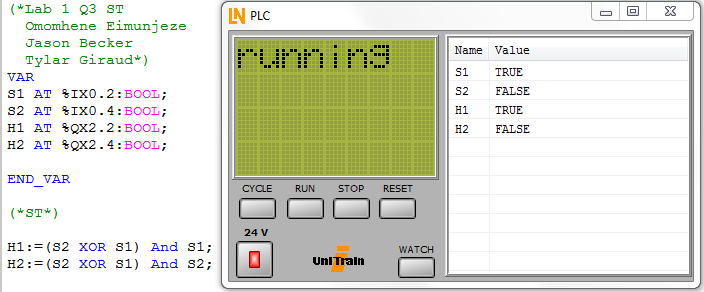


Image 7- Code and Results for Exercise 3

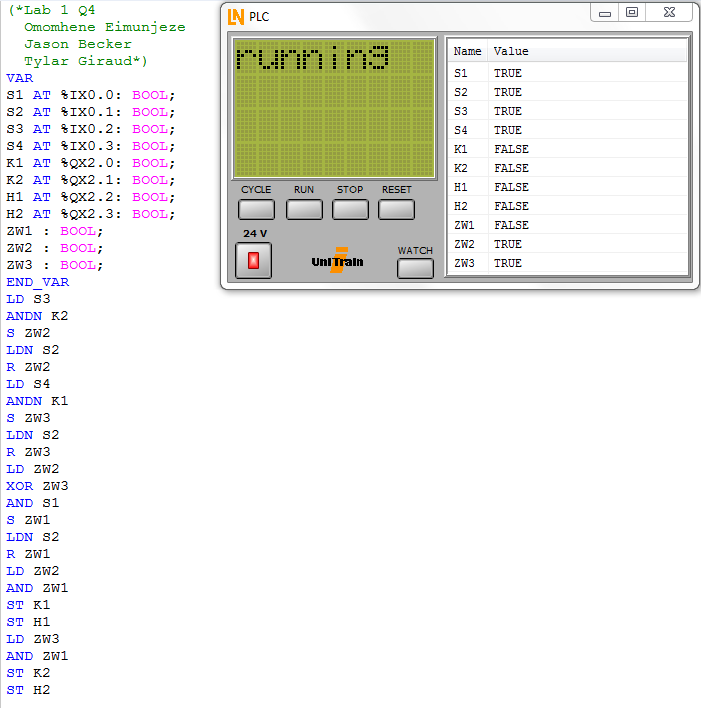
￼EXERCISE 4

Image 8- Code and results for Exercise 4

* **Program for Rotation Clockwise:**

If (S3 and NOT K2) is **True**: Contactor counterclockwise rotation K2 must be turned off when clockwise rotation S3 is actuated. However, K2 =1 when contactor counterclockwise rotation is actuated, therefore it should be negated ()

If (S3 and NOT K2) is **True**, then middle variable ZW2 should be True, that means we Set.

If (S2) is false then middle variable ZW2 should be False, that means we Reset and since the circuit diagram is a Reset Dominant Latch the output will be zero.

If middle variables (ZW1 AND ZW2) is true, contactor clockwise rotation is ON (K1=1) and Signal lamp clockwise rotation is also ON (H1=ON), otherwise K1 and H1 will be OFF.

* **Program for Rotation Counterclockwise:**

If (S4 and NOT K1) is **True**: Contactor clockwise rotation K1 must be turned off when counterclockwise rotation S4 is actuated. However, K2 =1 when contactor counterclockwise rotation is actuated, therefore it should be negated ()

If (S4 and NOT K1) is **True**, then middle variable ZW3 should be True, that means we Set.

If (S4) is false then middle variable ZW3 should be False, that means we Reset and since the circuit diagram is a Reset Dominant Latch the output will be zero.

If middle variables (ZW3 AND ZW1) is true, contactor counterclockwise rotation is ON (K2=1) and Signal lamp for counterclockwise rotation is also ON (H2=ON), otherwise K2 and H2 will be OFF.

* **Program for Middle Variables ZW2 and ZW3**

Since motor cannot run simultaneously clockwise and counterclockwise, that means when (S3 and NOT K2) is turned ON, (S3 and NOT K1) should be turned OFF. To perform such task, a logic XOR operator is required. Therefore, when one of the outputs either (ZW2 or ZW3) = ON AND S1 is ON, outputs of ZW1 will be ON until Reset button S2= =, which has a Reset Dominant Latch.

#### **Conclusion**

In conclusion, we see that the major differences between ST and IL coding. In this lab there were alot of logic gates and we learned how to code and utilize them in terms of examples. In each example, we see multiple gates being used and the variety of results, we can get from changing the inputs. The program MULTISIM proved useful for this lab as well since we can easily make truth tables and test our logic gates and coding by simply recreating the circuits and using the logic converter.

##### **References**

* LUCAS-NUELLE LABsoft, PLC and Bus Technology- Basic Boolean Logic, Combination of Boolean operations and Storing