

Connected Components Workbench

Task-based Quick Start

QS# GS09 – Making Runtime Changes

LISTEN.
THINK.
SOLVE.®

 *Allen-Bradley • Rockwell Software*

**Rockwell
Automation**

Copyright © 2009 Rockwell Automation, Inc.

Hardware & Software Versions Used to Develop This Quick Start

- CCW Release 1, Build 51
- 2080-LC30-16QWB v1.03

Prerequisite

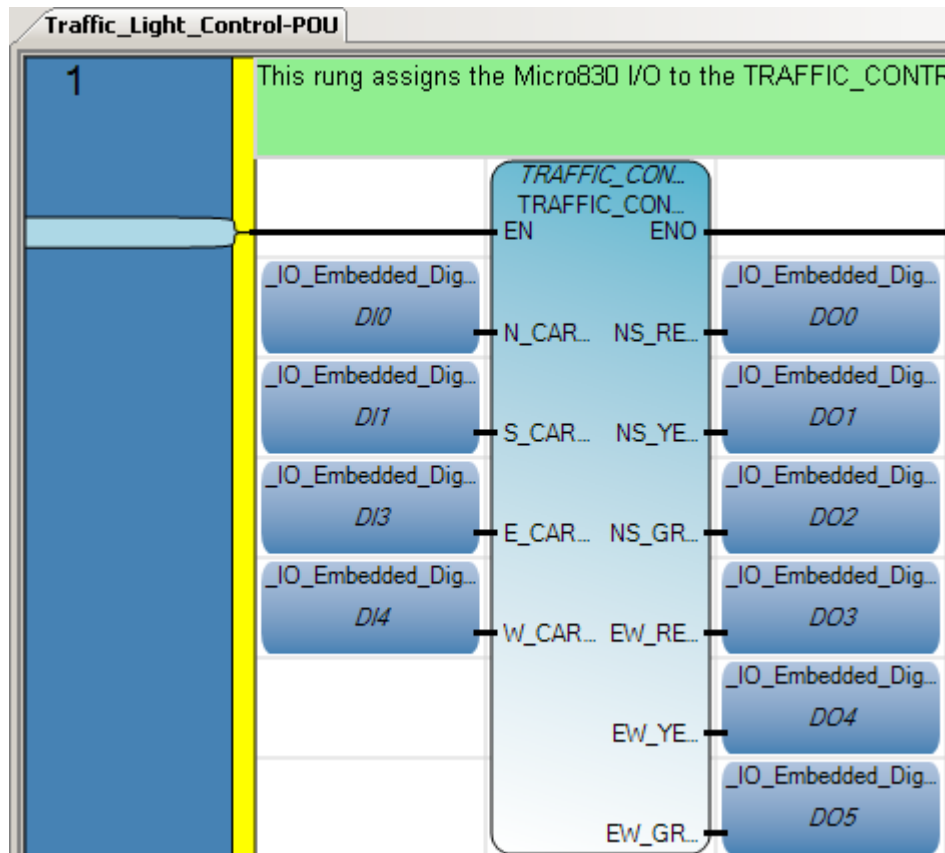
- QS# GS07 – Downloading a Project to a Controller

Making Runtime Changes

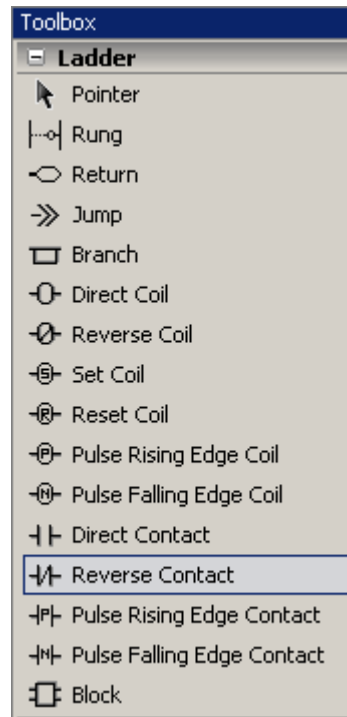
This quick start will show you how to make runtime program changes, which are changes to an existing program that are downloaded and implemented in a running controller (without going to program mode).

Note that this advanced functionality, also called Run-time Download, is only available in the Developer version of Connected Components Workbench software.

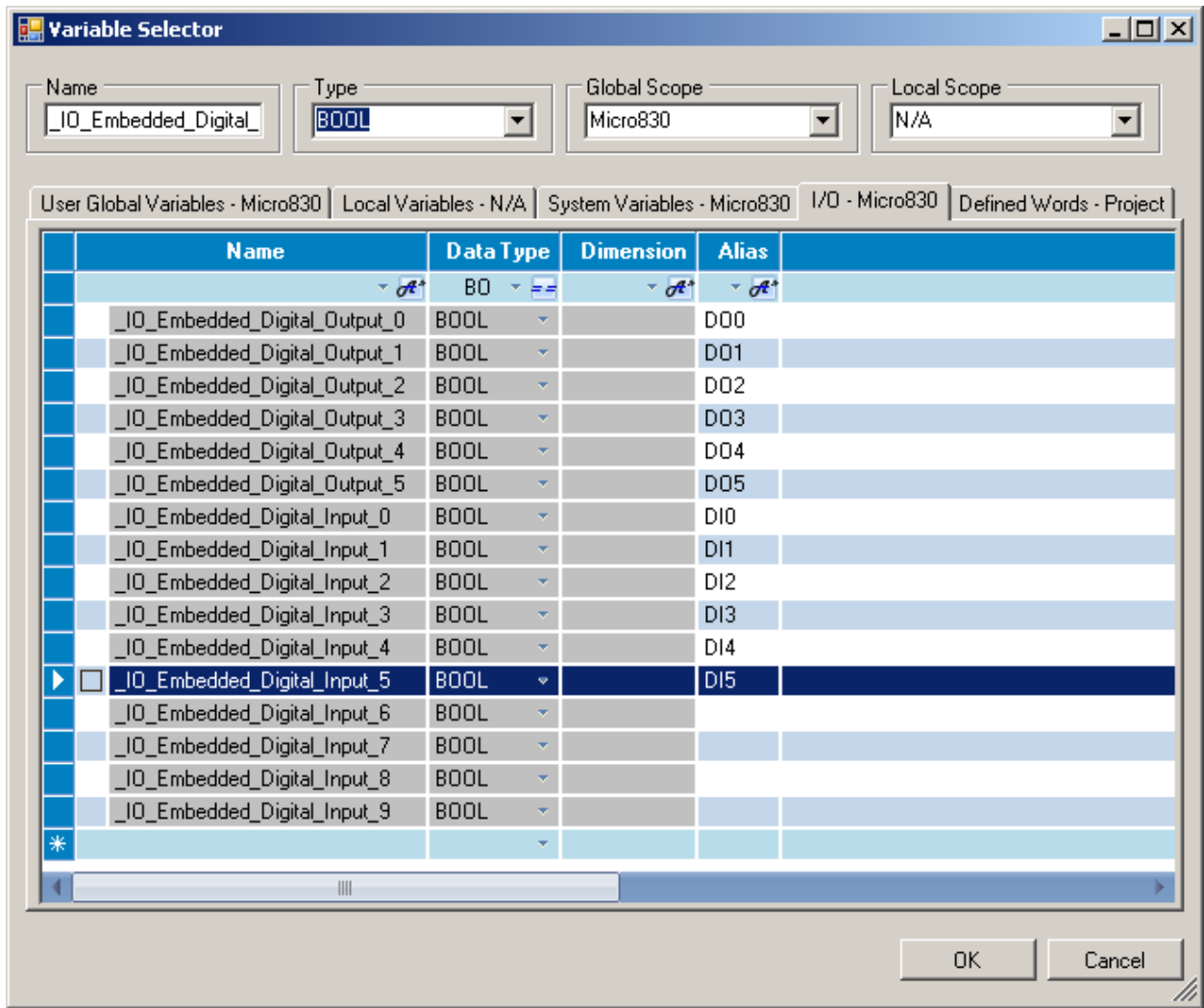
1. Starting with the one-rung Traffic_Light_Control program (which should be loaded and running in the controller), the next step is to make program modifications that will flash the lights based on an input change. The North/South lights will flash red and the East/West lights will flash yellow. However, we want to download these enhancements as Runtime Changes, meaning the controller remains in Run mode throughout the process.



- From the **Toolbox** in the right-hand screen, click and hold on **Reverse Contact**, drag it onto the rung between the left-hand rail and the TRAFFIC_CONTROLLER function block and release:

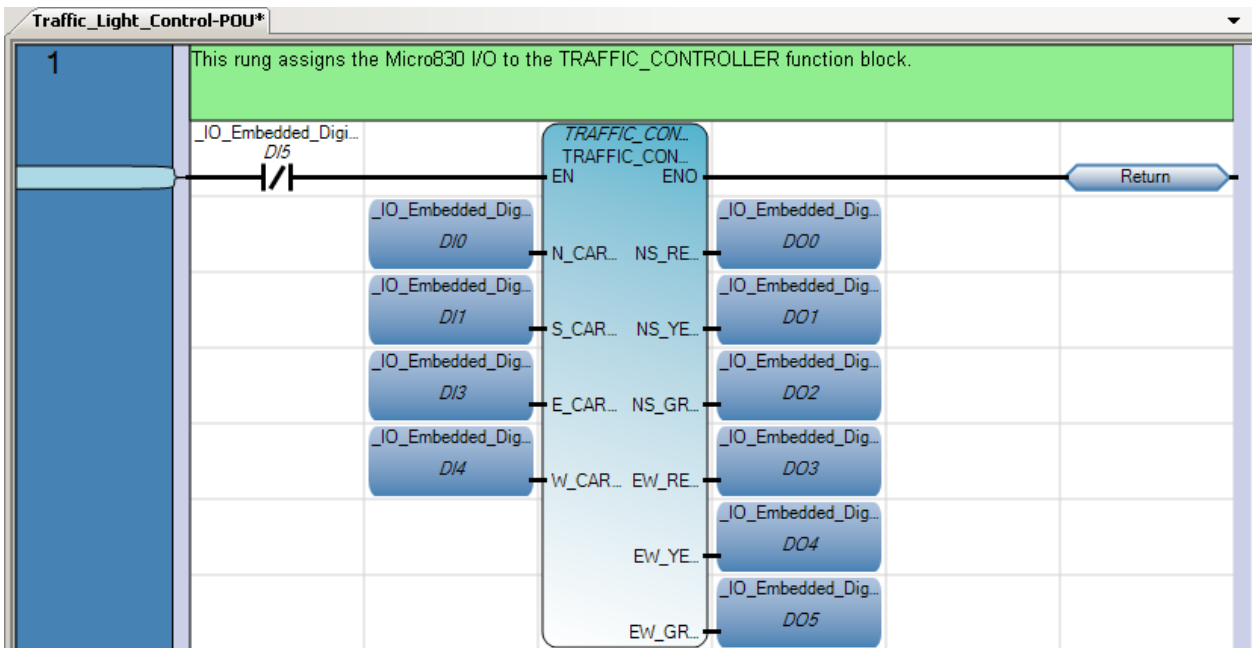


- When the **Variable Selector** screen appears, select **_IO_Embedded_Digital_Input_5** from the **I/O – Micro830** tab and click **OK**:

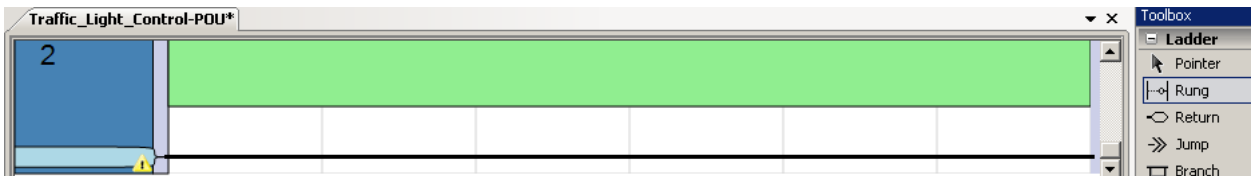


4. Next drag a **Return** from the **Toolbox** onto the rung to the right of the TRAFFIC_CONTROLLER function block and release. That way, if the TRAFFIC_CONTROLLER function block is enabled, no other rungs of ladder in this file will get executed. If the TRAFFIC_CONTROLLER function block is disabled, then the remaining rungs in this ladder program will get executed.

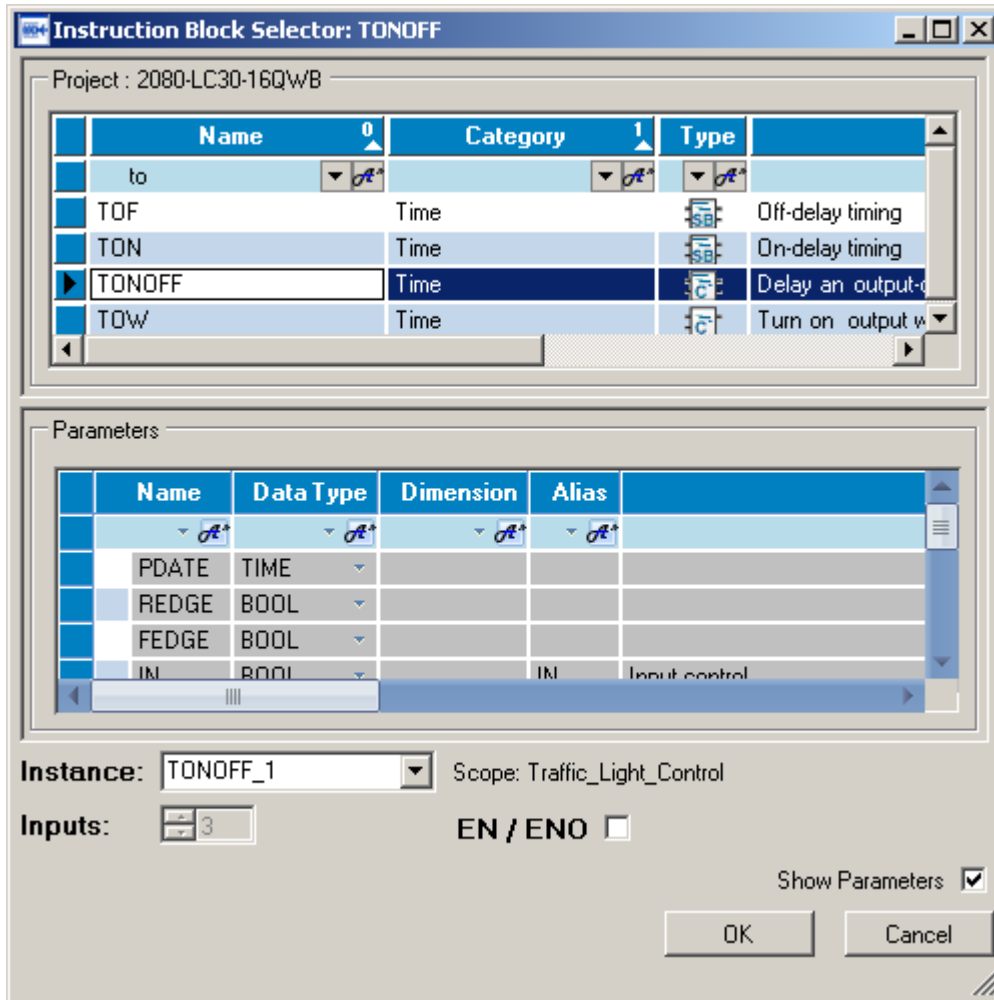
5. Your first rung should now look like this:



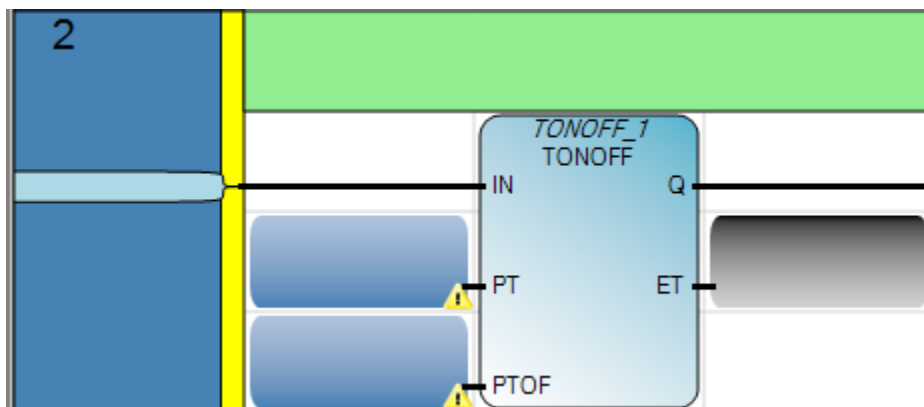
6. Next you are going to add a rung which will flash the traffic lights when the TRAFFIC_CONTROLLER function block is disabled (_IO_Embedded_Digital_Input_5 on). From the **Toolbox** in the right-hand screen, click and hold on **Rung** and drag it beneath the existing rung and release:



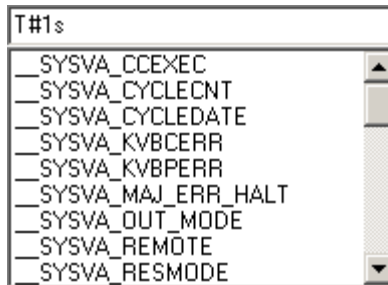
- Now add a 1 second on/1 second off timer to control the flashing of the red lights in the North/South direction and the yellow lights in the East/West direction. Drag a **Block** from the **Toolbox** and place it on the new rung. When the **Instruction Block Selector** screen appears, enter in "to" for the name, select **TONOFF** for Timer On-Off and click **OK**:



Your rung should look like this so far:

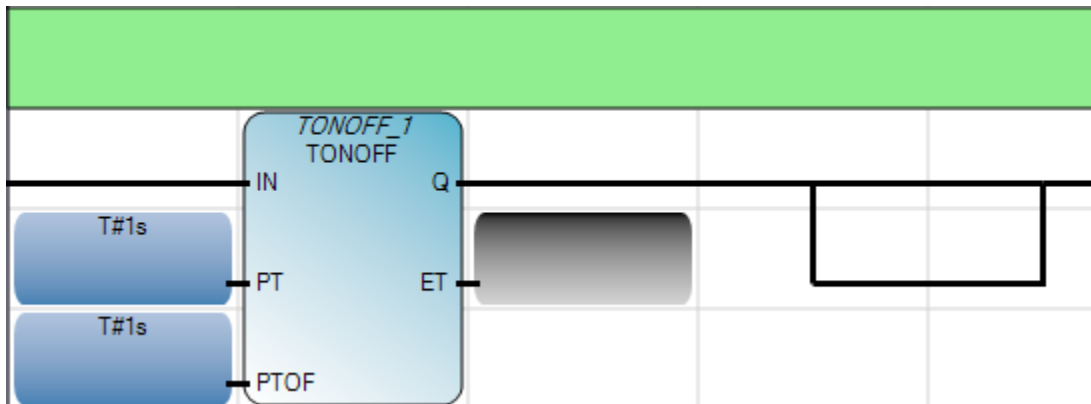


- As long as the Input (**IN**) to the **TONOFF** is true, then the Elapsed Time (**ET**) begins timing until the Elapsed Time equals the Preset Time (**PT**), at which time the Output (**Q**) becomes true. When the **IN** goes false, the **ET** resets and begins timing until the **ET** equals the Preset Time Off (**PTOF**), at which time the **Q** goes false. The caution triangles ⚠ next to **PT** and **PTOF** indicate that either a variable name or a constant value must be entered into these input blocks. We want this timer to be a 1-second on/1-second off timer, so click in the top of the **PT** input block and enter in “**T#1s**”, where **T#** means this is a Time constant and **1s** indicates 1 second:

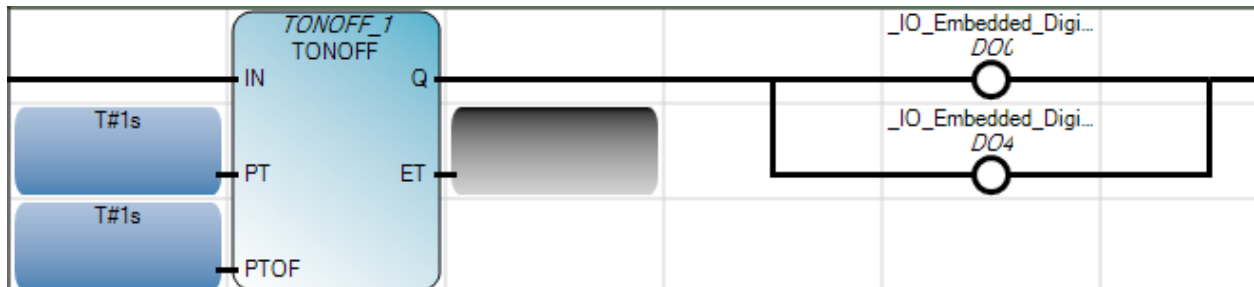


Repeat for the **PTOF** input block.

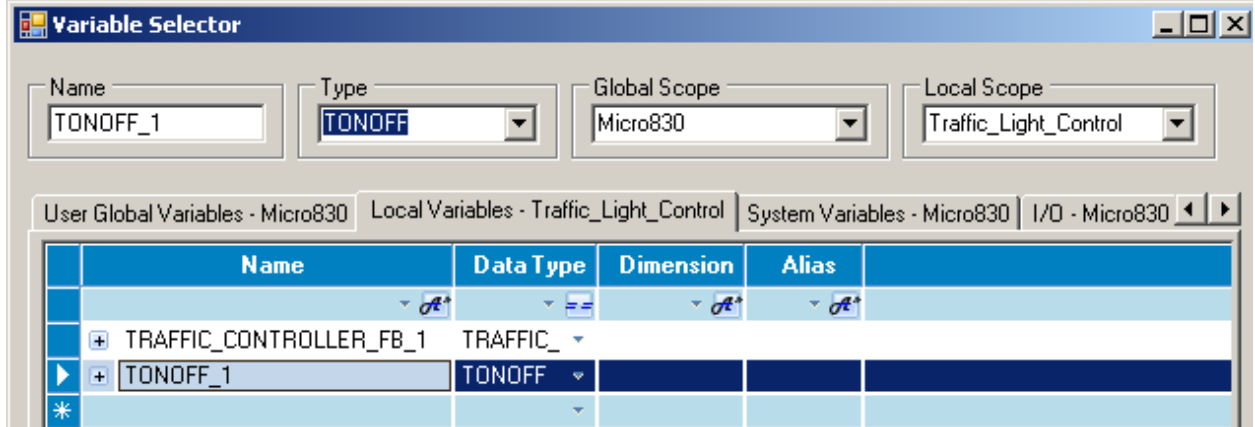
- When the on-timer has timed out and the output **Q** becomes true, we want the **NS_RED_LIGHTS** (DO0) to turn on, as well as the **EW_YELLOW_LIGHTS** (DO4). Since there are two outputs to be controlled off of **Q**, a parallel branch is needed on the rung. Drag a **Branch** from the **Toolbox** onto the rung just right of the **TONOFF**:



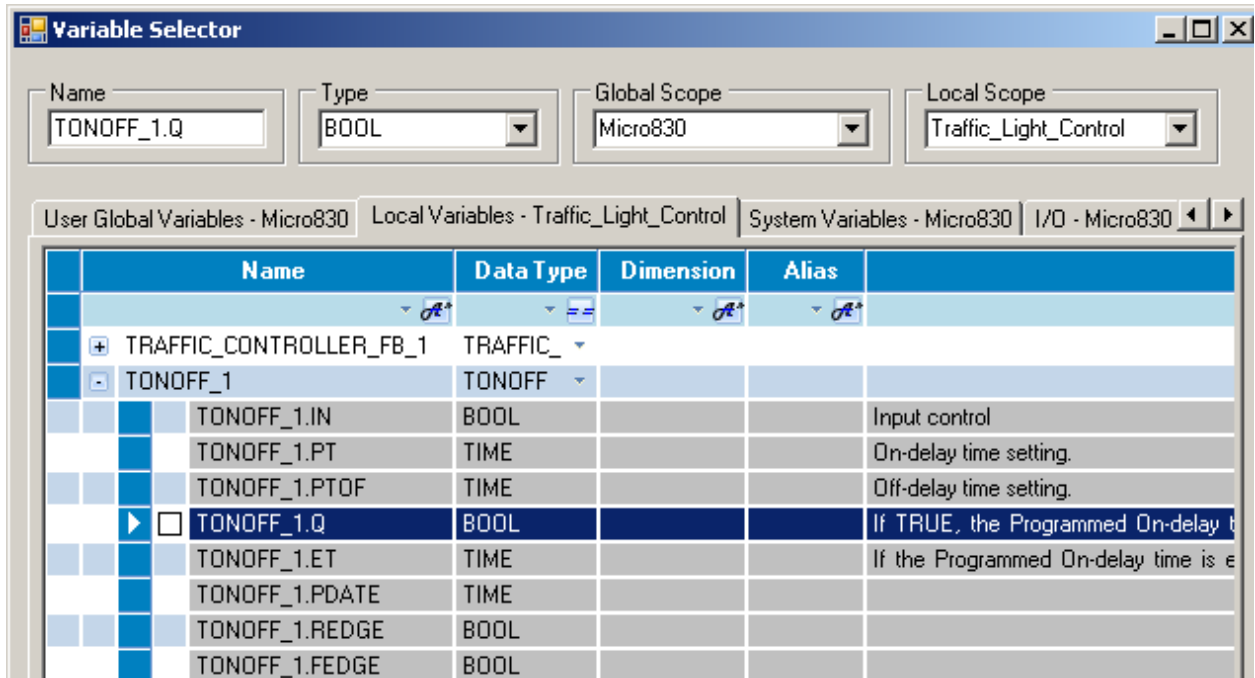
- Both DO0 and DO4 are to be controlled by **TONOFF** output **Q**, so drag a **Direct Coil** from the **Toolbox** to the upper branch and assign variable **_IO_Embedded_Digital_Output_0** to it and drag a second **Direct Coil** from the **Toolbox** to the lower branch and assign variable **_IO_Embedded_Digital_Output_4** to it:



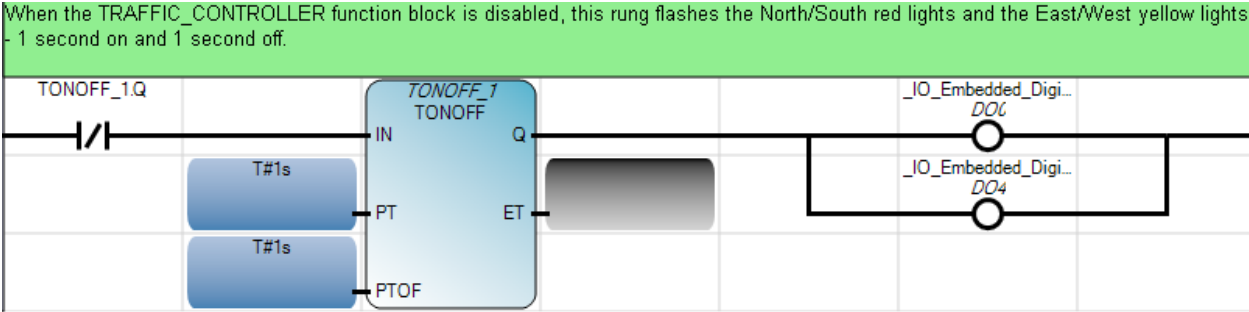
11. Next, **IN** needs to turn off once **Q** turns on in order to trigger the off timer. Therefore, you will add a **Reverse Contact** to the left of the **TONOFF IN** that is controlled by the **TONOFF Q** bit. Drag a **Reverse Contact** to the left of the **TONOFF** block. When the **Variable Selector** screen appears, click on the **Local Variables – Traffic_Light_Control** tab. If no names appear in the **Name** column, click on the cell to the right of the "*" and you should see the following:



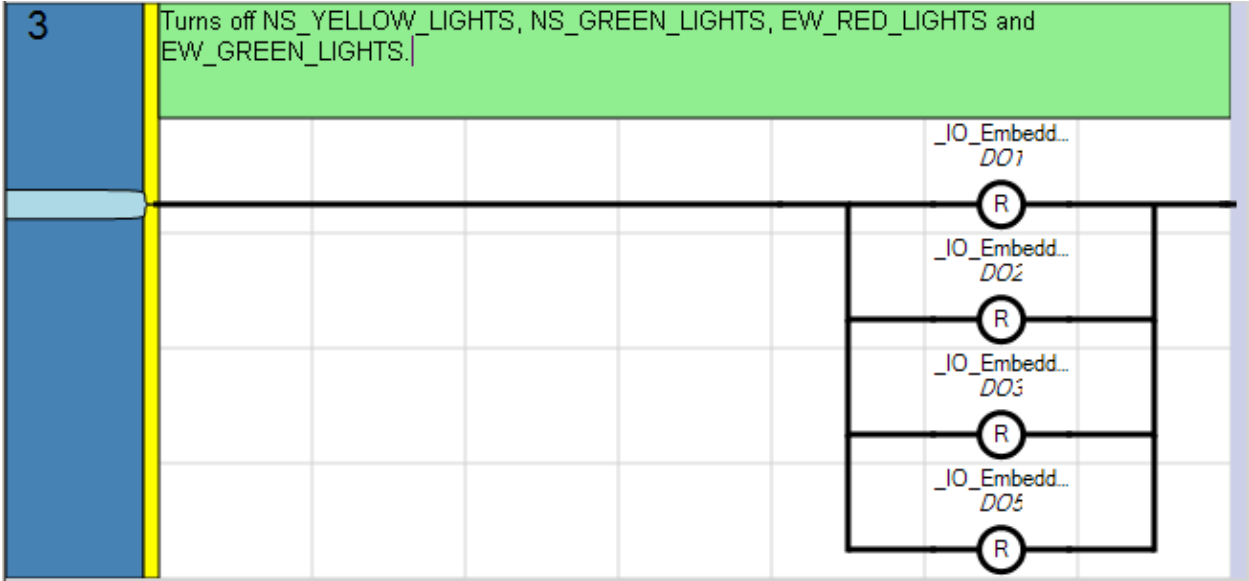
12. Click on the "+" to the left of **TONOFF_1** to expand its variables. Select **TONOFF_1.Q** and click **OK**:



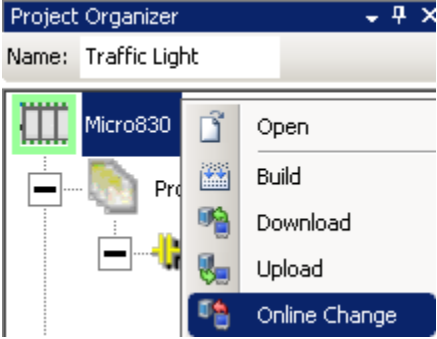
13. Document the rung by double clicking in the green space and typing “When the TRAFFIC_CONTROLLER function block is disabled, this rung flashes the North/South red lights and the East/West yellow lights – 1 second on and 1 second off.” Your rung 2 should look like this:



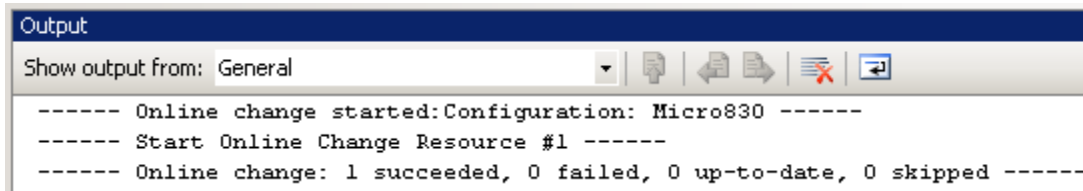
14. Finally, add a rung as shown that makes sure that the other four lights (controlled by DO1, DO2, DO3 and DO5) are turned off:



15. **Build** and **Save** the updated program. While connected to the controller, verify that the controller is still in **Run** mode. Instead of selecting **Download**, select **Online Change** or **Run-time Download** (which terminology is used depends on the release of software):



15. Verify in the **Output** window that the **Online Change** succeeded:



16. Test the program change out by turning embedded input 5 on and verifying that embedded outputs 0 and 4 flash on and off.