SY486K Connected Components Workbench v13 Installation Guide

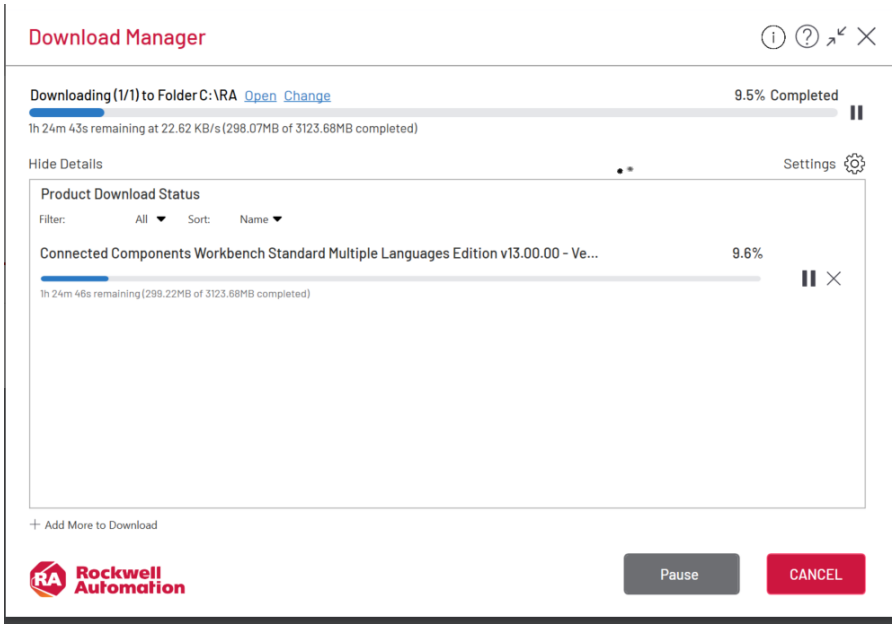
United States Naval Academy, Spring AY2023

Part 1: Downloading CCW

1. Create an account with Rockwell Automation
    a. Use your school email and put United States Naval Academy as for company
2. Use this link to access the CCW download
    a. Click on the plus sign next to the blue colored Connected Components Workbench to expand the versions.
    b. Next to 13.00.00, click 'Select Files.'
    c. Enter your login information, and you should be presented with a pop-up window with multiple select options.
        i. Select **Connected Components Workbench Standard Multiple Languages Edition v13.00.003.05 GB** under Configuration Software
        ii. Select the red shopping cart in the upper right hand corner of the pop-up window.
    d. Click Download Now
    e. At this point, if you did not receive a pop-up window with the title 'Rockwell Automation End User License Agreement', you may have to enable third-party cookies to receive the correct license agreement.
        i. Go to Google Settings → Privacy and security → Cookies and other site data
        ii. Click Add and enter the following two sites: [*.]rockwellautomation.com and https://compatibility.rockwellautomation.com:443
    f. Click 'Accept and Download' under the license agreement.

Part 2: Installing CCW/Configuring Windows Settings

1. After accepting the license agreement, you should get another pop-up with the download instructions.

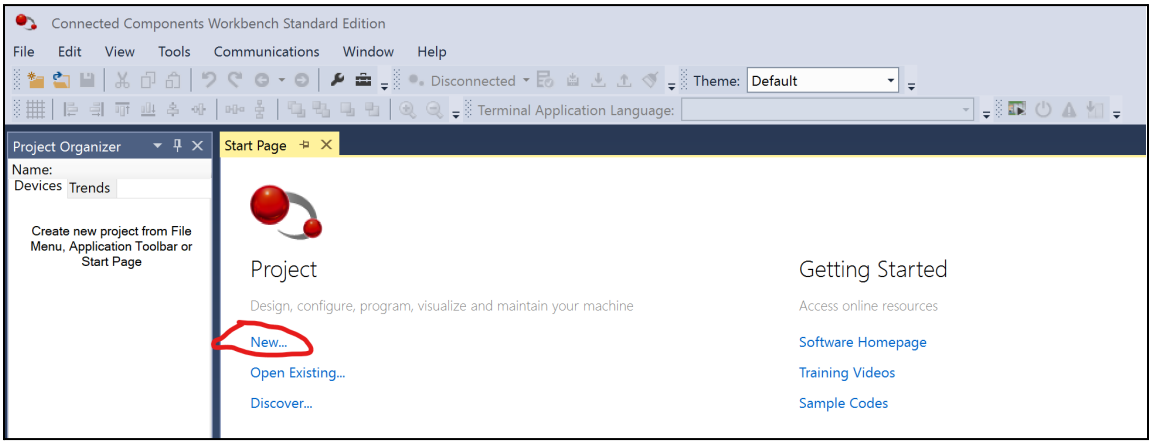


2. Do a normal download, (not direct download) TBD picture of that.
3. After the download completes, (if the wizard doesn't automatically launch) you should look for a new folder in your main system C: drive called 'RA' that contains the install program for CCW.
    a. Navigate to the folder OS (C:) → RA → 13.00.00-CCW-INT-Std-DVD in File Explorer
    b. Double click on the **CommonInstall.exe** file to execute the installation
    c. If you receive an error message stating that you need to install NET Framework 3.5, complete the following steps to configure Windows settings.
        i. Right-click Start, and click Run.
        ii. Type regedit.exe, and click OK
        iii. Go to the following registry key (you can also copy and paste this into the address bar at the top): HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU
        iv. On the right-pane, if the value named UseWUServer exists, set its data to 0
        v. Exit the Registry Editor
        vi. Restart Windows
        vii. Type Turn Windows features on or off in the search bar.
        viii. Enable NET Framework 3.5
        ix. If needed, install the files for the NET Framework 3.5
        x. Exit out of the window
    d. If required, also run the Setup.exe file

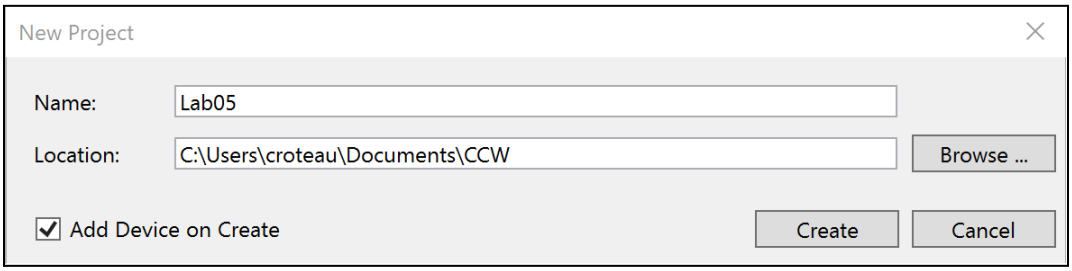
e. You should now be able to finish the CCW Install and complete the lab.

Part 3: Creating you first Ladder Logic Program

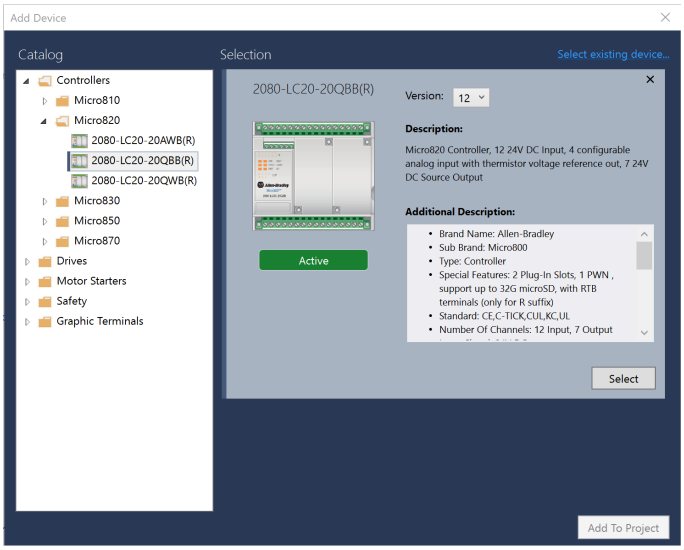
1. Read over the introductory material on this page: https://ladderlogicworld.com/ladder-logic-basics/
2. Then we are going to create some of the simple programs from this page https://ladderlogicworld.com/ladder-logic-programming-examples/
3. Poke around in the CCW program, check out the help menu
4. When you are ready to begin from the Start Page click on the "New…" project link



5. Give your project a catchy title like "Lab05" and change the location if you care to.



6. When the "Add Device" dialog pops up expand the Controllers and Micro820, then select the model that ends with "BB", hit "Select" then "Add to Project"



7. You should see an explorer tree on the left, under your project click on the Global Variables entry, it should appear blank. Next to the entries for Digital Output (*DO_01-4) # 1 through #4, give them names similar to what is shown below for how they are connected to the lights on the top row of the trainer devices. The furthest left light in DO1, and the red on the right end is DO4.

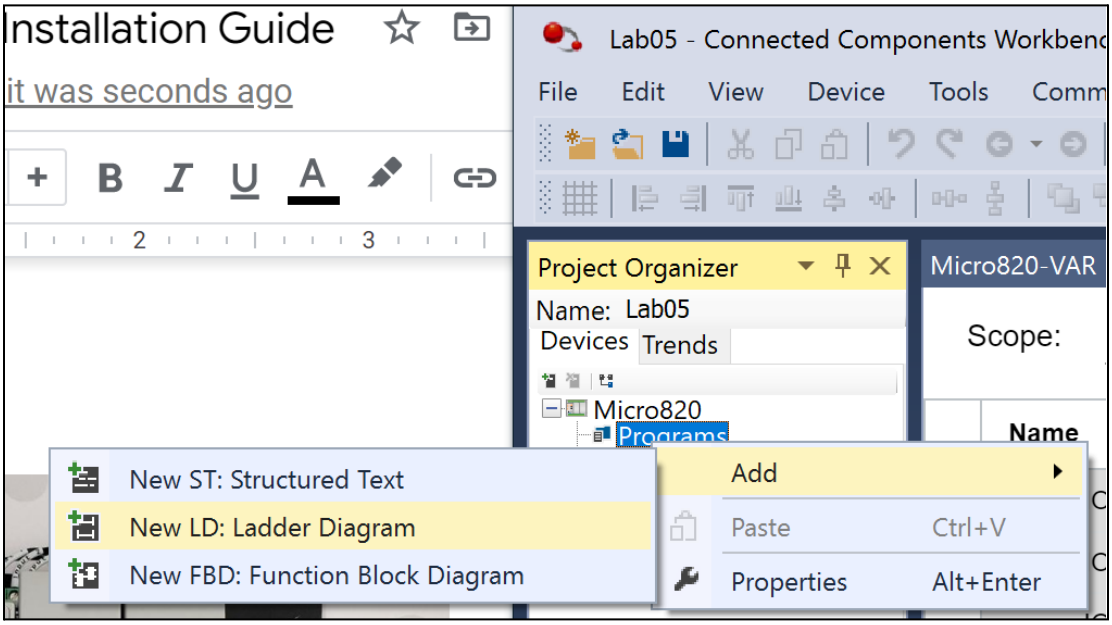
Micro820-VAR | Prog1-POU | Micro820 | Start Page

Scope: Micro820 Filter...

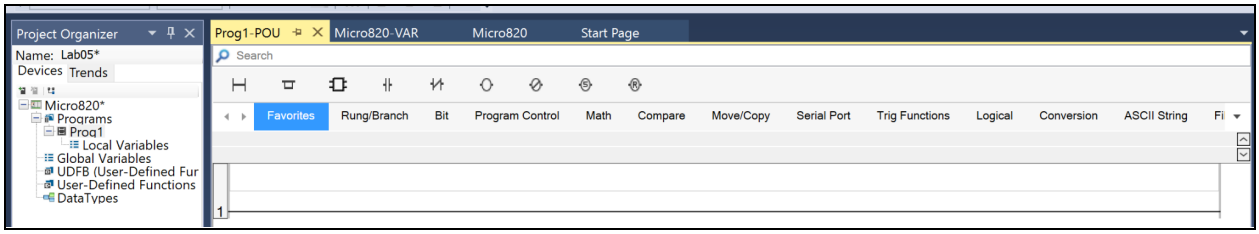
Name	Alias	Data Type	Dimension	Project Value
_IO_EM_DO_00		BOOL		FALSE
_IO_EM_DO_01	Blue_Light	BOOL		FALSE
_IO_EM_DO_02	Green_Light	BOOL		FALSE
_IO_EM_DO_03	Yellow_Light	BOOL		FALSE
_IO_EM_DO_04	Red_Light	BOOL		TRUE
_IO_EM_DO_05		BOOL		FALSE
_IO_EM_DO_06		BOOL		FALSE
_IO_EM_DI_00		BOOL		FALSE
_IO_EM_DI_01	ToggleSw1	BOOL		FALSE
_IO_EM_DI_02	ToggleSw2	BOOL		FALSE
_IO_EM_DI_03	PushButtonGreen	BOOL		FALSE
_IO_EM_DI_04	PushButtonRed	BOOL		TRUE
_IO_EM_DI_05		BOOL		FALSE

8. Do the same thing for the Digital Inputs (*DI_0?), two toggle switches and the two push buttons.

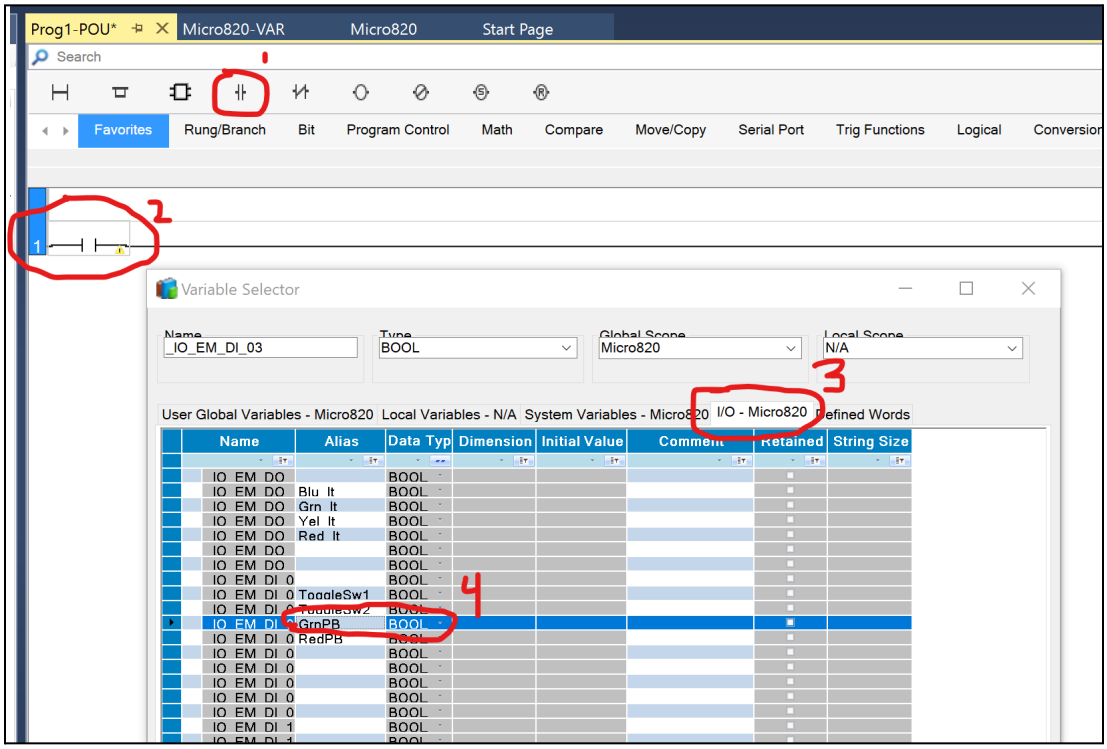
9. Now we are ready to create the ladder logic program itself. Right click on the explorer tree off to the left on the entry labeled "Programs" then select "Add", and "New LD: Ladder Diagram"



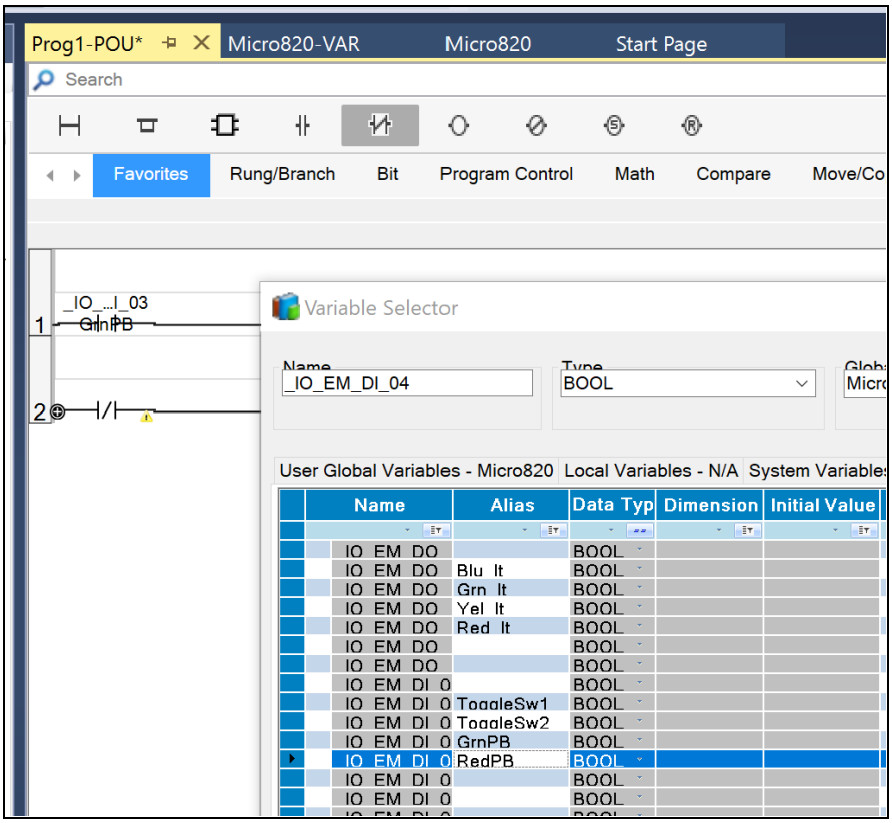
10. This will create a new entry called "Prog1" that you can now open by clicking on in the explorer tree.



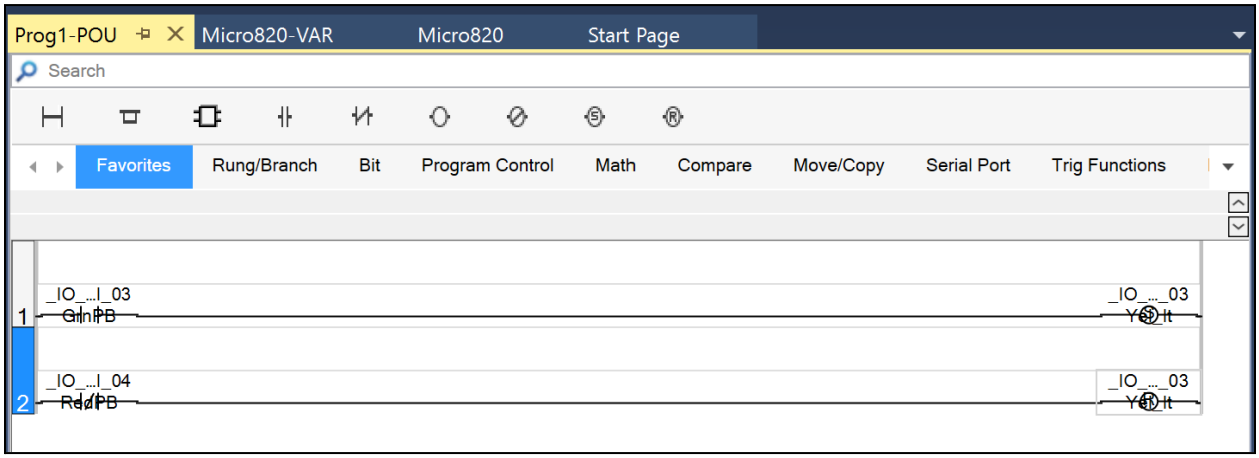
11. Let's make the first example of a motor which needs a start and stop button wired to it.
12. On the first rung which was created for us, drag down a Direct Contact (you might have to grab yours from the toolbox over on the right edge of the window) to the left edge of the rung. That will open up a Variable Selector to tie an input to that contact. Select the "I/O" tab near the right edge and look for the Green Push Button that you tied to "_IO_EM_DI_03" then click OK.



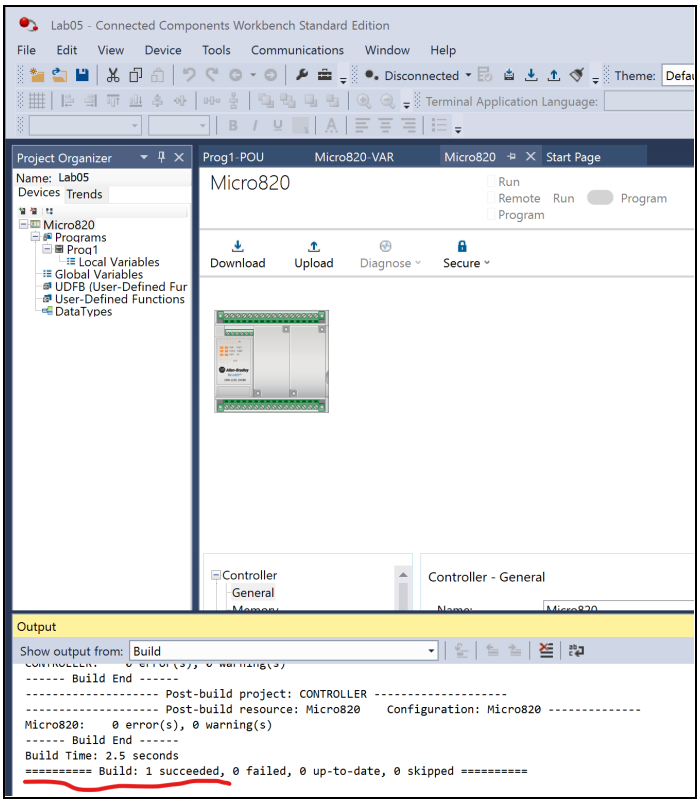
13. Next look for a "Set Coil" drag that anywhere on the rung to the right of the Contact (look for the black plus sign), next link that coil to the Yellow Light or "_IO_EM_D0_03" which is pretending to be our motor for this first exercise.
14. From there add another Rung below the first by dragging one out and placing it below the blue "1" numeral on the left rail.
15. After that grab a Reverse Contact, this is required because on the trainer kits we have the red pushbutton is a Normally Closed switch, instead of what the green one is Normally Open. And we want the yellow light to go out after we press the Red button so we have to "invert" the signal coming from the Red RB to only Reset when it is pressed.



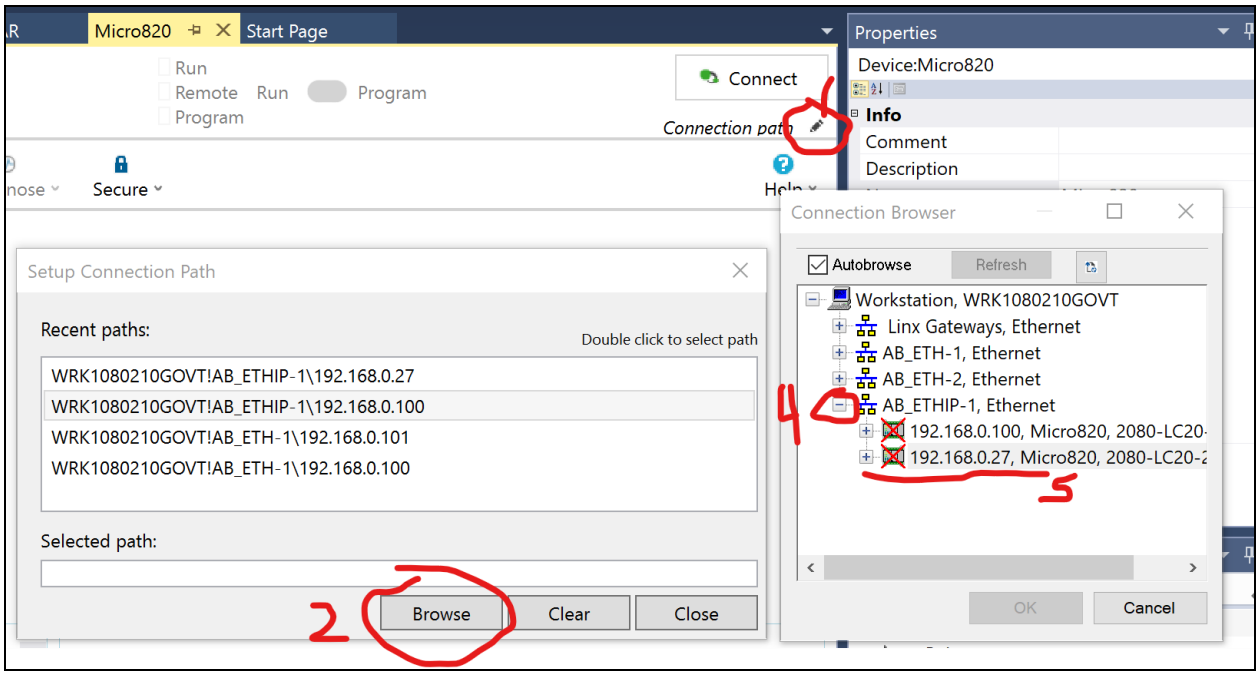
16. Then select the aforementioned "Reset Coil" and add that to the second rung and also tie it to the Yellow Light (DO 3).



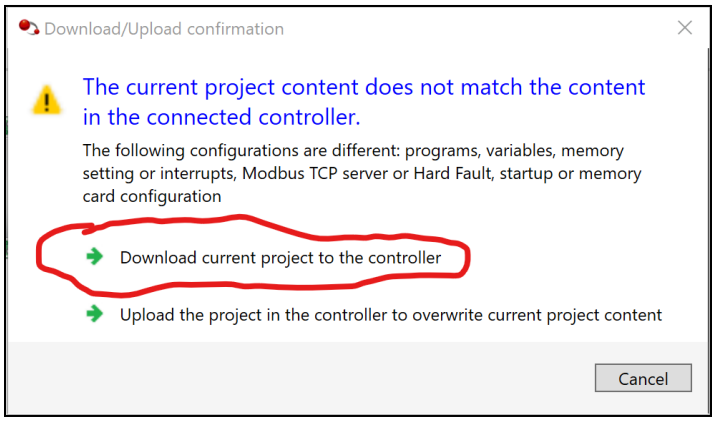
17. Save your project, then look for the Build icon in the toolbar, and look for any errors on the console window below. This should take you back to the Micro820 page of your project.



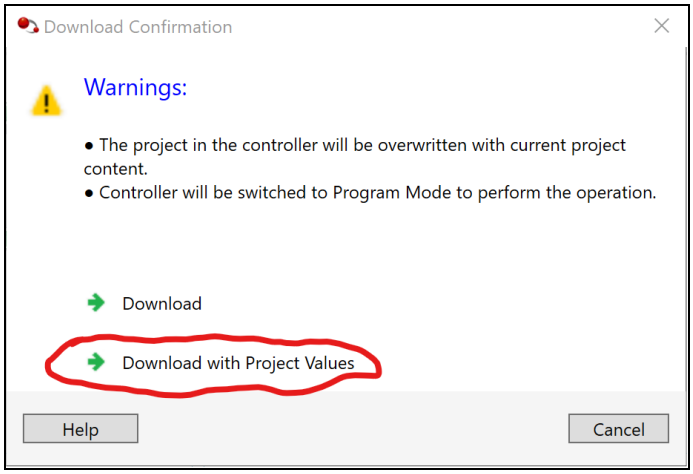
18. To connect to the PLC, plug in your laptop to the extra ethernet cable at your station. Make sure that you also turn off your wifi. You will still be able to access the internet from this "cyber forensics" network, but don't try to connect to two different networks at once.
19. Now it is time to connect to the PLC and DOWNLOAD your program to it. In the upper right corner of the IDE right below a "Connect" button look for a small pencil icon with the tool tip "Setup Connection Path." Or you can get there from the Device menu.

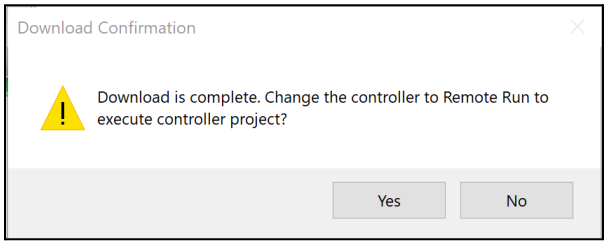


20. From there select the "Browse" button then open the connection with an "ETHIP" or something similar and look for the IP address of the PLC that is at your station. Hopefully yours won't be X'ed out and you should be able to select it, then select "Close."
21. Then once that IP address is shown in the upper right you can click the connect button and it should pop up a window warning you that the contents of the PLC doesn't match what is in your project, which is okay and what we expect. Remember once we connect the wording here is from the perspective of the PLC, so select the opinion to download the current project to the PLC and overwrite what is currently there.

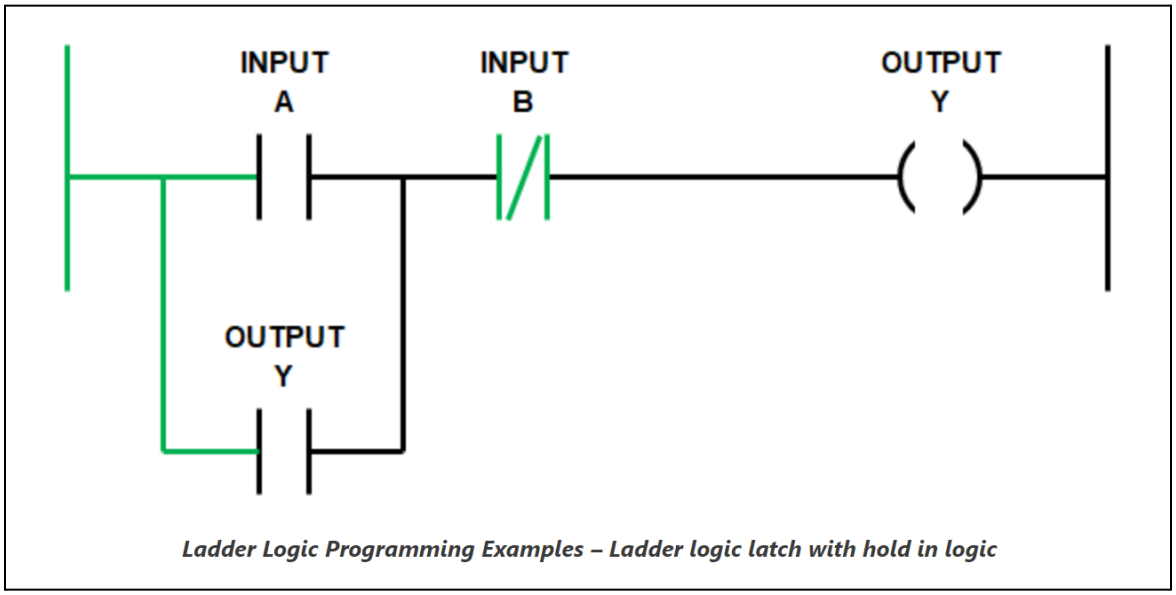


22. It might pop another window asking if you want to preserve the currently loaded local variables, but select the bottom option to load the variables from your project instead.
23. It should load your program into the PLC, you might see the switch in the IDE go from Run to Program. Then it usually asks you if you want to switch the PLC back into Run mode so you can run your program and you should answer in the affirmative.





24. That should be it. Your simple latch program should allow you to turn on and off the Yellow light with the green and red push buttons.
25. Again read through the discussion on this page about how the signals get passed on the ladder logic, remember the idea of ladder logic that things flow "top to bottom, left to right." While it is running, look at the ladder program in CCW and see things turn blue and red.
26. Not all PLCs have these fancy Set and Reset Coils, so after you save this Project, make a copy of it and change the logic to use the structure shown here. Try making this one to turn on and off the Blue light using the two toggle switches.



Ladder Logic Programming Examples – Ladder logic latch with hold in logic

27. The important thing to remember in our case is if you use the Red pushbutton as Input B you will have to invert the logic from what is shown here and make it a Normally Open or Direct Contact in AB's terms from the IDE.
28. If you are feeling further inspired look up a way to toggle on and off a single output using a single input.