Hardware Engineer:	Andrew Struthers	Lab:	6
Software Engineer:	Basil Alsulami	Due:	February 21st, 2019

Lab 6: Analog Data and Interfacing

Purpose:

This lab will use a low voltage, low current photovoltaic cell as an analog light sensor.

Objectives:

Upon completion of this lab, you should be able to:

- Use comparison instructions in conjunction with timers to sequence events.
- Interface analog sensors to the PLC
- Read data from the Analog module and use basic math operations to manipulate the analog data.

Materials:

1 *x* PLC Trainer
1 *x* Photovoltaic Cell
1 *x* Digital Multi-Meter
Fluke 179 or equivalent

Procedure:

Part A: Initial Software Setup

1. Before beginning any lab, always inspect the equipment to verify that all connections, including the chassis ground wires are secure and properly connected and that the equipment is safe to operate.

2. If necessary, use the **Network Connectivity and Configuration** procedure from Lab 2 to configure the network connection for your PLC from the PC that you plan to use for the remainder of this lab.

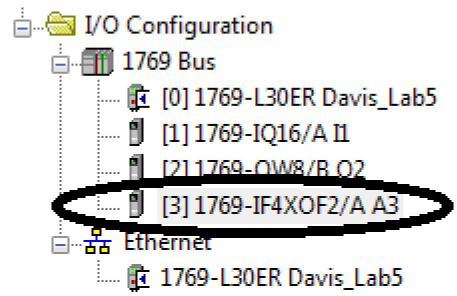
3. Launch **Studio 5000** and create a new Project named "Lab6_Unit#" where # corresponds to your Unit number. **Save the file to your N: drive.**

Note:
If a program template containing the default IO configuration, program tags, and the Machine_Active rung has been created, skip to step 7.

4. Configure the Controller I/O Configuration as outlined in Lab 2.

5. Configure the Program Tags using the same tag labels as used outlined in Lab 3.

6. Right click on the 1769-IF4XOF2/A A3 module under the I/O Configuration folder and select “Properties”.

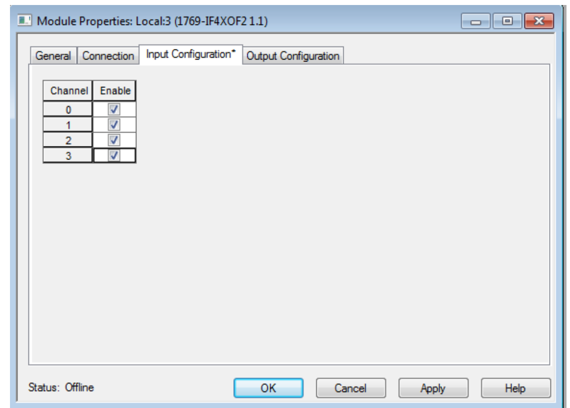


7. Open the **Input Configuration** Tab and verify that all channels are enabled. Verify that all output channels are likewise enabled in the **Output Configuration** Tab.

8. Implement the **Machine_Active** rung as described in Lab 4 on the **MainRoutine** programming pane.

9. Create a rung that will enable the **White_Indicator** only if **Machine_Active**=0 on the **MainRoutine** programming pane.

10. Save your project before continuing and go offline if applicable.

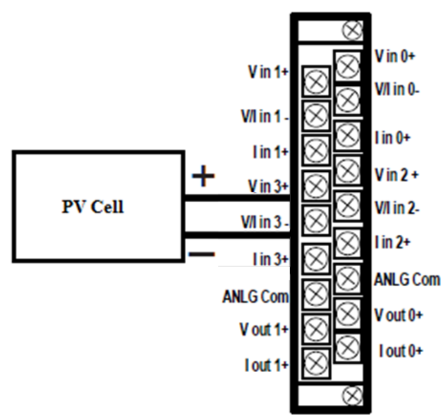


Initial Hardware Setup:

The following procedure will describe how to interface the photovoltaic cell to one of the input channels on the analog module of the PLC.

11. Verify that the power is turned off to the unit.

12. Route a brown wire from the **V in 3+** terminal and a white wire from the **V/I in 3-** terminal on the 1769-IF4XOF2 analog module to two available terminal blocks on the upper terminal block rail.



Note:

The illustration to the right does not show the terminal block connections.

13. Use a DMM to verify the polarity of the leads connected to the photovoltaic cell. The red plug should be the positive lead and the black plug should be the negative lead.

14. Wire the red and black banana jacks on the right side of the cabinet to the terminal blocks where you made the connections in step 12. Make sure that the red jack is connected to the terminal block associated with **V in 3+** and the black jack is connected to the terminal block associated with **V/I in 3-**.

15. Connect the PV cell banana plugs to the jacks on the cabinet. Verify that the terminals of the PV cell are isolated from the case of the PLC unit and that all connections are securely connected before proceeding.

16. Turn the power to the unit on. After the PLC has completed its self-test, measure the terminal voltage on the PV cell. This module should be producing between 5 to 8 V. Make sure when measuring the voltage that the PV Cell is facing up and is exposed to ambient light.

17. Open the project for this lab in **Studio 5000** and go online.

18. Open the **Controller Tags** tab and record the **Local:3:I.Ch3Data** value in the first line of comments.

⊟ Local:3:I	{...}	{...}	
⊞ Local:3:I.Fault	2#0000...		Binary
⊞ Local:3:I.Ch0Data	0		Decimal
⊞ Local:3:I.Ch1Data	0		Decimal
⊞ Local:3:I.Ch2Data	0		Decimal
⊞ Local:3:I.Ch3Data	$\boxed{0}$		Decimal
⊞ Local:3:I.InputRangeFlag	2#0000...		Binary

19. Partially block the light to the PV Cell. The value stored in **Local:3:I.Ch3Data** should change. If it does not, verify that the electrical connections are secure and that the controller is Online.

20. Recall that the data for the module is stored in bits 7 through 14 of the sixteen bit word. Create a **Program Tag** of type **INT** called **Channel3_Input_Data**. Use a math operation such as **DIV** or a bit shift register such as **BSL** to move the data into the lower seven bits of the program tag created above.

Word	Bit Position															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	SGN	Analog Input Data Channel 0								0	0	0	0	0	0	0
1	SGN	Analog Input Data Channel 1								0	0	0	0	0	0	0
2	SGN	Analog Input Data Channel 2								0	0	0	0	0	0	0
3	SGN	Analog Input Data Channel 3								0	0	0	0	0	0	0

21. Create a **Program Tag** of type **REAL** called **Voltage_Input**. Use a multiplication operation to calculate the Input Voltage based on the value stored in **Channel3_Input_Data**.

    Note: Observe that the input channels have a range of 0 – 10 V and a resolution of 2^8 bits. Therefore, the current resolution is:

$$\frac{Voltage}{Step} = \frac{10\ V}{256\ steps} = 39.0625\frac{mV}{step}$$

22. Download the program to the PLC and go online. Record the calculated voltage in the space provided.

Calculated Voltage
7.4 V

23. Measure the terminal voltage on the PV cell with a DMM and record the value in the space provided. The calculated voltage should be within 10% of the measured voltage before continuing.

Measured Voltage
7.8 V

Part B

1. Use the program created in *Part A* to write a program that will perform the following operations.

    - Calculate the power delivered by the PV Cell.
        - Note: For this activity, assume that the internal resistance of the module is 100 Ω. The power can be calculated by:

$$P = \frac{V^2}{R} = \frac{V^2}{100}$$

    - Turn on the **Red_Indicator** only when $\mathbf{Voltage\_Input} < \mathbf{6\,V}$. This will represent a low light condition.
    - Turn on the **Yellow_Indicator** only when $\mathbf{6\,\mathit{V} \leq Voltage\_Input} < \mathbf{9\,V}$. This will represent a nominal light condition.
    - Turn on the **Green_Indicator** only when $\mathbf{Voltage\_Input} \geq \mathbf{9\,V}$. This will represent a high light condition.

Note:

It may be necessary to modify the control limits based on the ambient light in the room and the output voltage of your PV Cell.

2. Verify that you have saved your project and that the program is well-documented before proceeding.

3. Download the program to the Controller and test its operation. If an unexpected behavior occurs, troubleshoot the hardware or software to resolve the issue.

4. Demonstrate **proper** operation of the project to the Instructor or a TA to receive a sign-off.

For Submission:

Students are required to submit a PDF copy of the program for Part B. Include the recorded value from Step 18 and your revised control limits in the first line of comments. The program should be well-documented.