Experiment M12
PLC Thermostat
Procedure

Deliverables: checked lab notebook, tech memo, demonstration of working device to Lab TA

Overview

A Programmable Logic Controller (PLC) is a microcontroller or computer used to automate an industrial process. The PLC and other supporting components are typically mounted on a DIN rail, which is a standard mount for industrial automation equipment. (DIN is an acronym for Deutsche Institut für Normung.)

Shown in Fig. 1 below is a photo of a DC power supply, PLC, and terminal blocks mounted to a DIN rail. In this lab, you will construct the set-up pictured below and use it to make a thermostat. The PLC thermostat will use a thermistor to sense when a heated sample warms up to a desired high temperature set-point T_{HS}, then switch on a fan to cool the sample down to a low set-point temperature T_{LS}, at which the fans will be switched off.

This will all be very similar to what you have done with the Arduino UNO and breadboard in previous labs. However, the DIN rail mount and screw terminal electrical connections are far more reliable and robust.

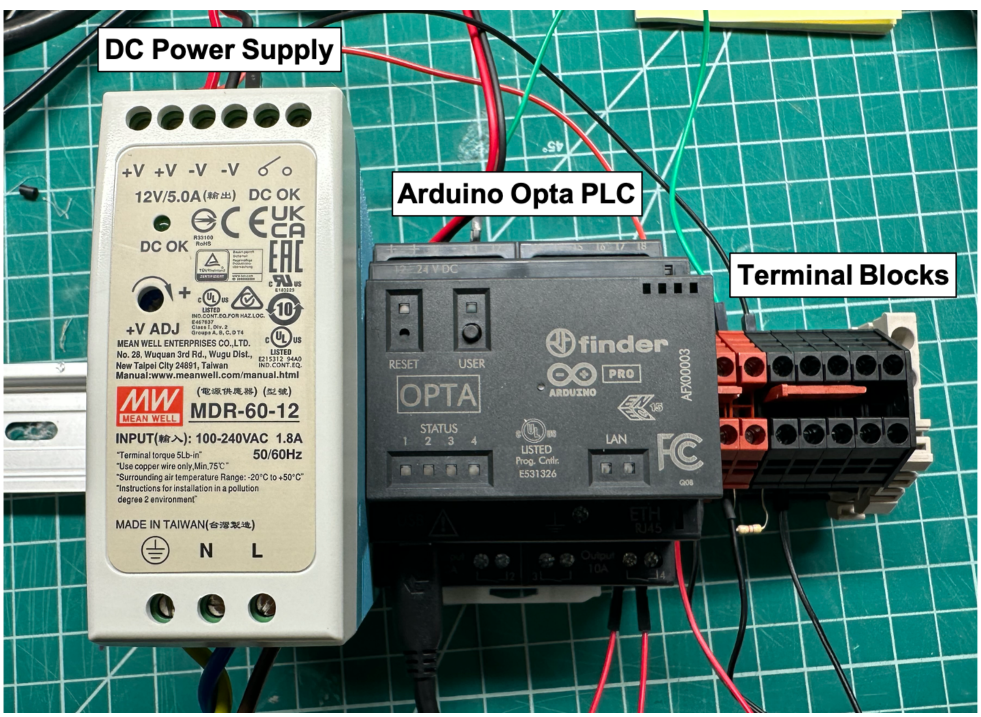


Figure 1 – A DC power supply, Arduino Opta PLC, and terminal blocks are mounted to a DIN rail and used to regulate the temperature of a heated sample.

Subsystem A: Opta PLC and DC Power Supply

You will begin by mounting the Arduino Opta PLC and 12V DC power supply onto a DIN rail. Then, you will install the Opta board drivers into the Arduino IDE and run a test program to verify it works.

Procedure

1. Mount the 12V DC power supply and Opta PLC to the DIN rail, as pictured in Fig. 1.
2. The +V and -V screw terminals on top output a differential voltage of 12V DC. Connect two lengths of 20 gauge wire to the +V and -V screw terminals at the top of DC power supply.

CAUTION: The N and L terminals at the bottom of the DC supply are connected to 120V AC. Touching these screw terminals will result in electrical shock.

Pro-Tip: For professional grade wiring, use the hex crimp tool to cap the stranded wire ends with ferrules.

3. Plug in the DC power supply, and green LED should light up. Use the DMM to verify that +V and -V output a 12V potential difference.
4. Unplug the DC power supply to turn it OFF. Connect +V and -V to the + and – power input terminals on the Opta PLC.
5. Open the Arduino IDE software, and install the Opta board compiler.
    a. Go to Tools > Board: > Boards Manager…
    b. Search for "Opta".
    c. Install the "Arduino Mbed OS Opta Board", if it is not already installed.
6. Plug in the DC power supply to power up the Opta PLC. Connect it to the computer using a USB C cable.
7. In the Arduino IDE, set the board to be the Opta.
    a. Go to Tools > Board: > Arduino Mbed OS Opta > Opta.
    b. Go to Tools > Target Core and make sure the Main Core is selected.
    c. Go to Tools > Port and select the port that the Opta is connected to.
8. Use an internet browser to access the Opta PLC user manual. Go to https://docs.arduino.cc/tutorials/opta/user-manual/
9. Scroll down to the User Button section. Read this section, copy the code, run it, and test it.
10. **Demonstrate the working push-button code to the TA, so you can be awarded points on your score sheet.**

Subsystem B: Blower Fan and PLC Relay

The Opta PLC contains built-in relays that can be opened and closed via the digitalWrite() function. You will now use one of these digital relays to cycle the blower fan ON and OFF.

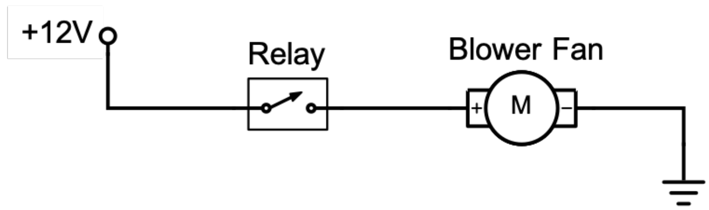


Figure 2 – A relay on the Opta PLC is used to automatically switch the blower fan ON and OFF.

Procedure

1. Test that the blower fan works by connecting it directly to 12V using male dupont pin jumper wires. Simply touch the wire pins to the -V and +V screws on the DC power supply.
2. Attach several terminal block to the DIN rail. Appendix B shows how to properly attach and remove them. Note the following:
    a. The front and back screw terminal of each block are electrically connected.
    b. Shorting jumpers can be used to connect adjacent blocks.
3. On the top side of the terminal blocks:
    a. Wire up one of the red terminal blocks to +V. (This will be +12V.)
    b. Wire up one of the black terminal blocks to -V. (This will be ground.)
4. Use the terminal blocks to construct the relay circuit in Fig. 2. The relays are located on the bottom of the Opta PLC.
5. Download the Fan Template code from the lab webpage.
    a. Replace the first *** with the correct digital pin number corresponding to the relay. Refer to the Pinout in the Opta user manual for the digital pin numbers corresponding to each relay (D0 – D3).
    b. Replace the other ***s with correct values of HIGH, LOW, time durations, etc.
6. Upload the code to the Opta PLC. Every 3 seconds, the relay should energize and *click*, and the fan should switch ON or OFF.
7. **Demonstrate the working system to the TA, so you can be awarded points on your score sheet.**
8. Leave the circuit intact for the later parts of the lab.

Subsystem B: Thermistor

A thermistor will be used to measure the temperature of the sample. The thermistor will be wired up in the same voltage divider circuit you used in the C1 – C3 labs. The PLC will use its analog-to-digital convertor (A/D) to read the voltage output V_{out} from this circuit.

Procedure

1. The thermistor and a heater are already mounted to the aluminum plate. The heater wires are red and the thermistor wires are black. Use the other 12V DC power supply—the one that looks like laptop charger—to power the heater. Connect the red heater wires to it using the same kill switch and screw terminal adapter you have used in previous labs.
2. Wire up the thermistor in the same voltage divider circuit you used in the C1 – C3 labs. You may first construct a test circuit using the breadboard, but you will ultimately need to have everything wired up via the DIN rail terminal blocks.
3. Use the DMM to verify the thermistor circuit is working properly.
4. Use your code from the C3 lab to read in the thermistor circuit voltage and compute the temperature in degrees Kelvin. Note the following:
    a. The calibration constants for the new thermistor are A = 0.003361 1/K and

       B = 0.0002568 1/K.
    b. The Opta PLC has a 12 bit analog-to-digital converter. Enable the 12 bit A/D by entering analogReadResolution(12); in the setup. This means that 12V will be mapped to integer values between 0 and 4095 (instead of the 0 – 1023 that you are used to).
    c. Connect the thermistor circuit voltage output to any of input terminals on the top of the Opta PLC. Refer to the Programmable Inputs section of the Opta user manual for details and pinout.
    d. The program should print the time (seconds) and temperature (Kelvin) to the serial monitor.
5. Test the program by turning ON the heater with the kill switch. The temperature should rise when the heater is ON. Turn the heater OFF when you are finished.
6. **Demonstrate the working system to the TA, so you can be awarded points on your score sheet.**
7. Leave the circuit intact for the later parts of the lab.

Design Challenge 1 – Complete Thermostat System

You will now integrate Subsystems A, B, and C to create the complete thermostat system. Your thermostat should have the following features:

- Like any good thermostat, it should have a high set-point T_{HS}, and low set-point T_{LS}. The code should do the following with these:
  - Turn ON the fan if T > T_{HS}.
  - Turn OFF the fan if T < T_{LS}.
- In the setup portion of the code, use the single set-point T_S and hysteresis ΔT to calculate T_{HS} and T_{LS}, such that $T_{HS} = T_S + \Delta T/2$, and $T_{LS} = T_S - \Delta T/2$.
- The code should print the time, measured temperature, low set-point, and high set-point to the serial monitor, separated by commas.
- Test the program again with T_S = 324 K and ΔT = 2 K.
- **Demonstrate the working system to the TA, so you can be awarded points on your score sheet.**
- **Collect a temperature vs. time data set showing the oscillation between the two set-points. You will need to make a plot of this data for the tech memo.**

Design Challenge 2 – User Control

You will now use an analog potentiometer to allow a user to adjust the set-point T_S while the program is running.

- Wire up the top and bottom potentiometer wires to 12V and GND. Use a DMM to verify that the output voltage from the middle wire can be adjusted by turning the knob. You may first try using the breadboard, but the system must ultimate be wired up using only the terminal blocks.
- Use analogRead() to read in the potentiometer voltage into one of the inputs at the top of the Opta PLC.
- Map the range of potentiometer readings to a reasonable range of temperature set-points.
- As the user turns the potentiometer, the high and low set-points printed to the serial monitor should change.

Demonstrate the working system to the TA, so you can be awarded points on your score sheet.

Design Challenge 3 – Arduino PLC IDE

PLCs are typically programmed in a more sophisticated integrated development environment (IDE) using various graphical programming languages, such as ladder logic, functional block diagrams, structured text, sequential functions charts, and instruction lists. Arduino has recently released its own PLC IDE.

Use the Arduino PLC IDE software you installed on your laptop to upload and run *any* program you like on the Opta PLC (even a simple blink program).

- Refer to the following tutorial to set-up the Opta PLC with the PLC IDE software: https://docs.arduino.cc/software/plc-ide/tutorials/plc-ide-setup-license/
- You only need to use the PLC IDE to upload and run a simple blink program, but you can make this as sophisticated as you like.

Demonstrate the working system to the TA, so you can be awarded points on your score sheet.

Clean-up

To receive full credit, you must return the lab bench to its initial state:

- Disassemble the circuit. Disconnect the wires from the DC power supplies, Opta PLC, and terminal blocks.
- Use a flathead screw driver to pop the terminal blocks from the DIN rail. (See Appendix B for a photo of the proper technique.) Remove the DC power supply and Opta PLC from the DIN rail.
- Disconnect the USB cable from the computer.

Data Analysis and Deliverables – Write a brief tech memo containing the following item.

1. A plot for temperature vs. time for one of the tests you performed on the working thermostat.
   a. Include horizontal dashed lines denoting T_{HS} and T_{LS}.
   b. The caption should include the values of T_S, ΔT, and the period of the oscillations.

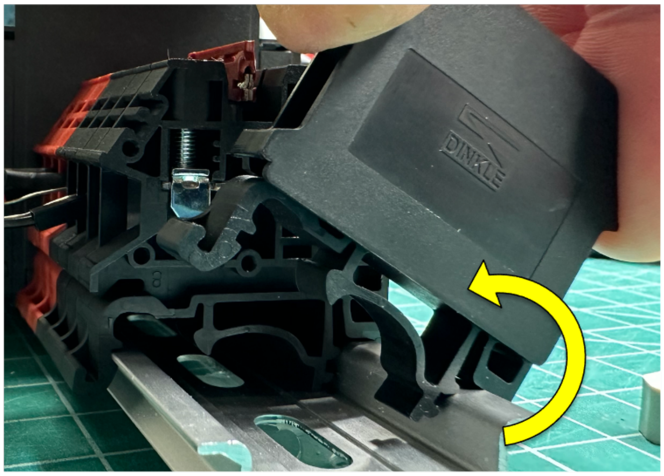
Appendix A

Equipment

- 12V DC power supply w/ inline kill switch and screw terminal adapter
- Arduino UNO Microcontroller
- 6ft USB cable
- Blower fan mounted to small plywood sheet – Amazon Part # B08P1S5DBN
- Breadboard mounted to small plywood sheet
- 2” × 2.5” × 1/16” aluminum sample w/ polyimide film heater and thermistor
    - Polyimide film heater – Amazon Part # B09X16XCVS
    - 10k NTC Thermistor, adhesive mount, Amphenol – Digikey Part # 235-1457-ND
- N-Channel MOSFET TO-220AB (Digikey Part # 497-2765-5-ND)
- Male-Male DuPont pin jumper wires
- 3M Velcro command strips

Appendix B

Attach Terminal Block



Remove Terminal Block