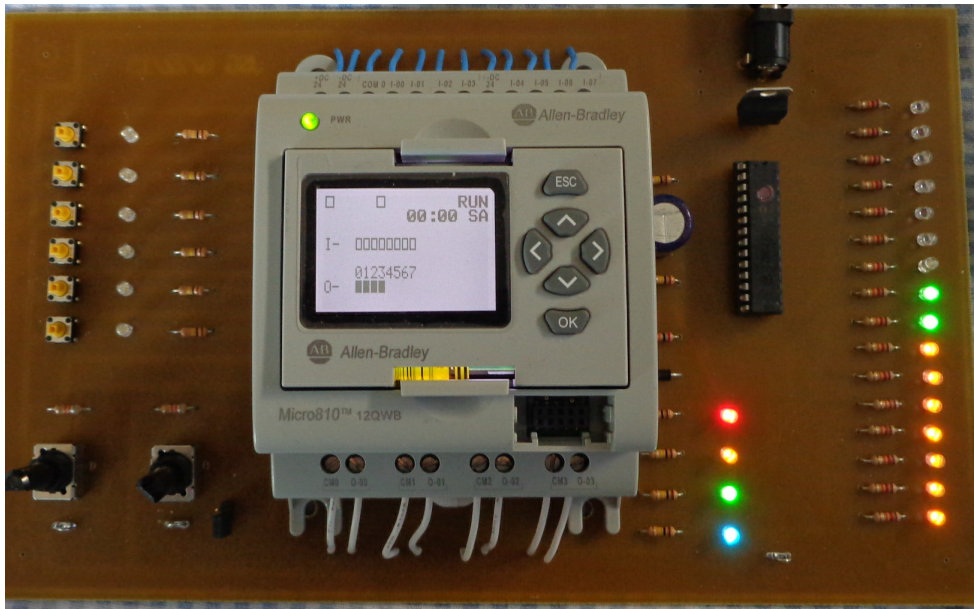

Programmable Logic Controllers



PLC Evaluation Board used in ECE 461

What is a PLC?

Programmable Logic Controllers (PLC's) are microprocessor devices much like the PIC microcontroller. Their function is to control the operation of systems such as a chemical processing plants. For this, they have several input ports. These measure the status of the outside world, such as recording when a car enters a parking lot or the temperature of a room. They also have several output ports. These serve to control valves, heaters, motors, or other devices.

Most controllers used by industry are PLC's. Some of the reasons for this are that

- PLC's are off-the-shelf controllers which require no custom designs
- Should a PLC fail, it can be replaced with another off-the-shelf PLC resulting in minimal down time
- PLC's are programmed using a symbolic language called "ladder logic." This allows the operator to change the program as nee be. Microprocessor-based controllers, on the other hand, are more difficult to program as anyone who has had an assembler course can tell you.
- PLC's are programmed. Changing the function of the controller simply requires you to change the program. Hardware controllers are more difficult and expensive to change since they would require a new hardware design.

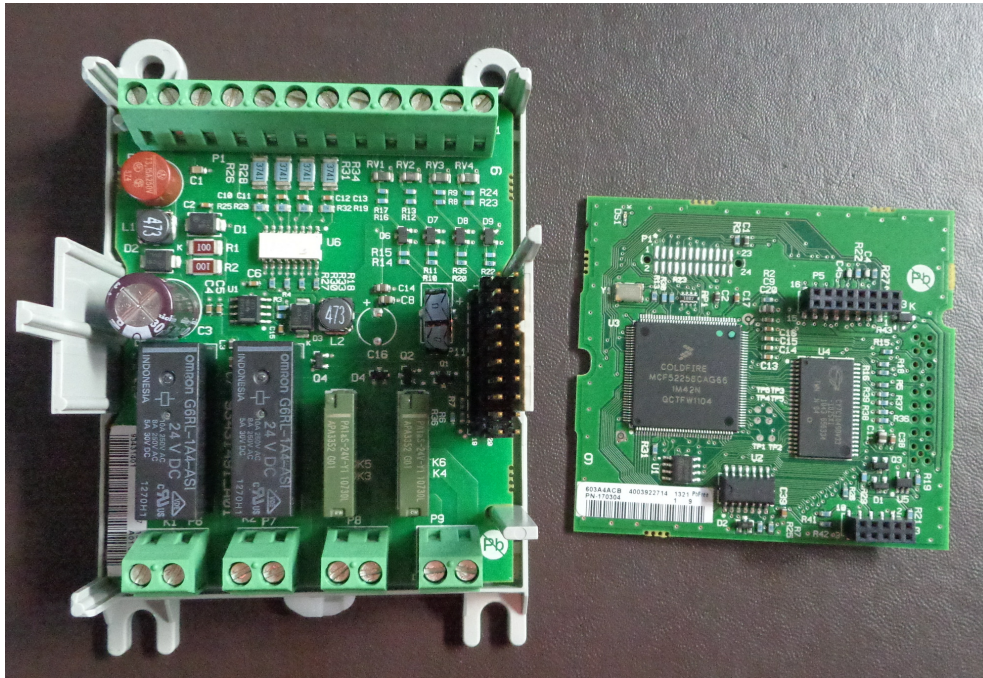
Likewise, the following four labs serve to introduce the students to what PLC's are, how they operate, and what some of the functions are.

Micro810 PLC

The PLC we are using in class is an Allen Bradley Micro810 model 12QWB. This version has

- 8 Binary Inputs (0V / 24V)
- 4 Analog Inputs (0V .. 10V)
- 4 Relay Outputs, capable of up to 3A for output 0 and 1, 1A for output 2 and 3

Internally, the PLC is just a microcontroller with some buffering on the inputs and relays for outputs.

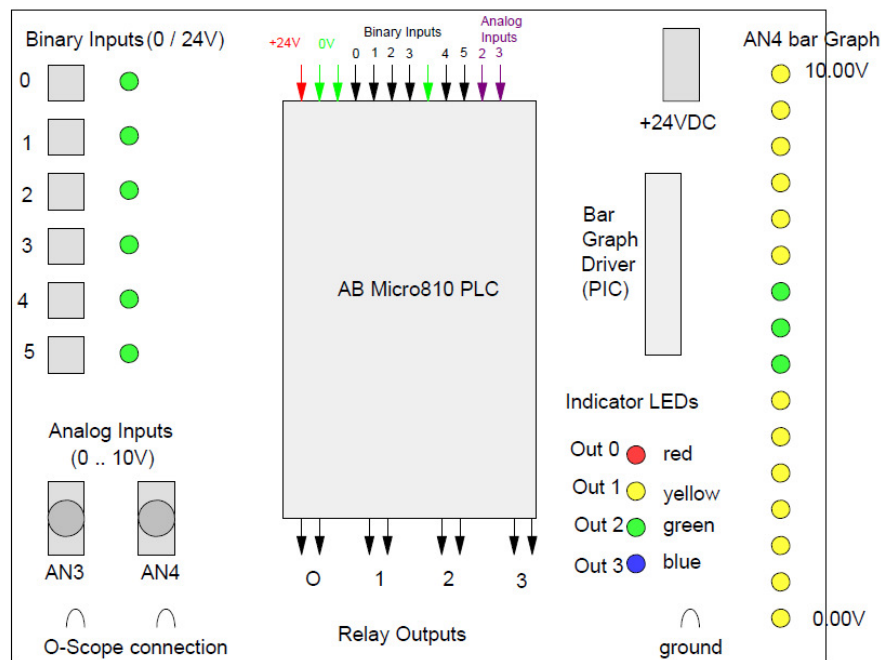


Guts of a PLC. The I/O board is on the left with buffering for the 8 inputs and relays for the 4 outputs. The microcontroller is on the right, which mounts above the I/O board.

Evaluation Boards used at NDSU

To make the Micro810 PLC easier to use, an evaluation board is provided. This connects the PLC to

- 24VDC through a power plug,
- 6 push buttons for binary inputs on Digital Input 0 .. 5
- 2 potentiometers to provide 0..10V to Analog Input 2 and 3, and
- LEDs to indicate when logic one is applied to the inputs or the relays are closed.



Evaluation Board Layout

The push-buttons on the left provide binary inputs to IN0 .. IN6

- Open = 0V (logic 0)
- Pressed = +24V (logic 1)

These allow you to simulate things like a person pressing a 'walk' button at an intersection or a car entering a car wash.

The two potentiometers at the bottom of the board provide two analog inputs to AN3 .. AN4. The voltage ranges from 0V (CCW) to +10V (CW). Two jumpers are provided as well in case you want to measure the actual voltage applied to the PLC.

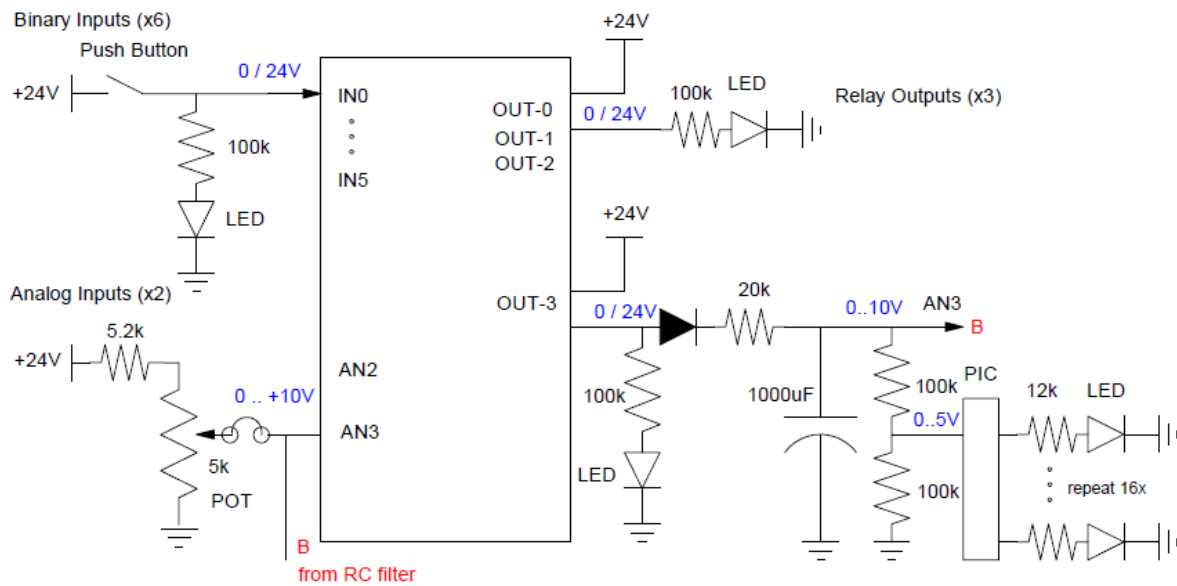
The outputs of the PLC are four relays (OUT0 .. OUT3). These are connected to four LEDs which indicate when the relays are closed (on) and open (off). They are also colored for simulating things like a stop-light.

The right side of the display is a bar graph with 16 LEDs. These indicate the voltage applied to AN4 from 0.00V to +10.00V in 16 steps. Three of the LEDs are green - allowing you to simulate a target for controlling the water level in a tank, or something like that.

OUT3 drives an RC filter so that

- When OUT3 is closed, the capacitor (and AN3) charges up to +10.0V
- When OUT3 is open, the capacitor (and AN3) discharges down to 0.0V.

The schematic is fairly simple and is as follows:



Evaluation Board Schematics

On the input side (left)

- When you press an input button on the left, +24V is applied to a binary input (IN0 .. IN5)
- When you turn a potentiometer, 0..10V is applied to the analog inputs (AN2, AN3)

On the output side (right)

- When a relay closes (OUT0 .. OUT3), +24V is applied to an LED and it turns on.
- In addition, when OUT3 closes, it charges up a 1000uF capacitor to +10.0V. This 0..10V signal drives a PIC processor, which drives a bar graph indicating the voltage, as well as analog input AN3.

Note that there is a jumper on analog input AN3.

- When the jumper is in place, the 5k pot drives analog input AN3 (it also charges the capacitor).
- When the jumper is removed, output OUT3 drives the RC filter as well as analog input AN3.
- Add the jumper if you want to control the analog input (simulating a sensor or an operator input).
- Remove the jumper if you want the RC filter to simulate something, such as the temperature in a room, the water level in a tank, etc.

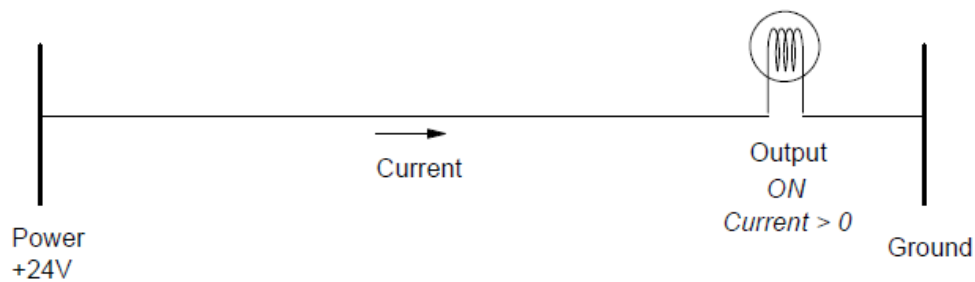
Ladder Logic

Ladder logic is a graphical way to implement a program. The notation is as follows:

- On the left side of the screen is a power bus.
- On the right side is ground.
- If you connect an output from power to ground, it turns on.

"On" in this case means

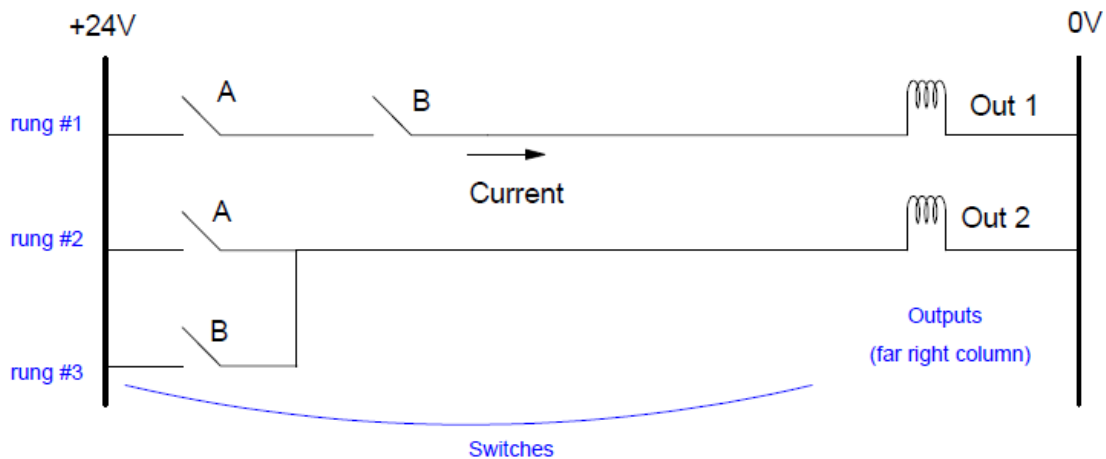
- The variable is logic '1' if it's an internal memory location,
- The output relay is closed if it's an output pin



Ladder Logic: When an output is connected to the left power rail, the relay is closed (on)

You can represent AND and OR gates by adding switches in series and parallel. For example, in the following diagram,

- Output #1 is ON if switch A and B are closed.
- Output #2 is ON if switch A or B is closed



Ladder Logic notation for an AND function (run #1) and OR function (rung #2)

Each line of a ladder logic program is called a rung.

Anything appearing before the last column is interpreted as a switch:

- Logic 1 means the switch is closed,
- Logic 0 means the switch is open

Anything in the last column is interpreted as an output relay

- Logic 1 means the relay is closed. If you connect a light from power to ground through the relay, the light turns on at logic 1
- Logic 0 means the relay is open (current = 0).

Ladder logic came out before graphical displays. As a result, keyboard symbols are used to represent relays and switches as follows:

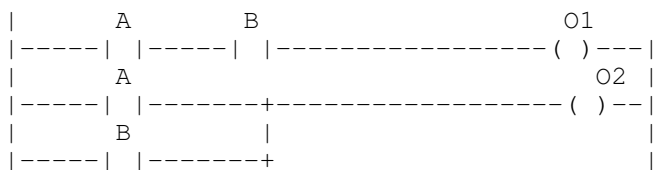
() A relay (must be in the last column)

|| A switch that is normally open (cannot be in the last column)

|/| A normally closed switch. Can also be interpreted as "not X"

Outputs can be used as a switch by placing that variable on the left. If used in this way, it means "if output X is true and"

For example, the above ladder logic program would actually look like the following on a computer with a character display



Ladder Logic Do's and Don'ts

- All outputs belong at the top of the program.
- Each output should appear only once.