Basic PLC program for control of a three-phase AC motor (for beginners)

Google+

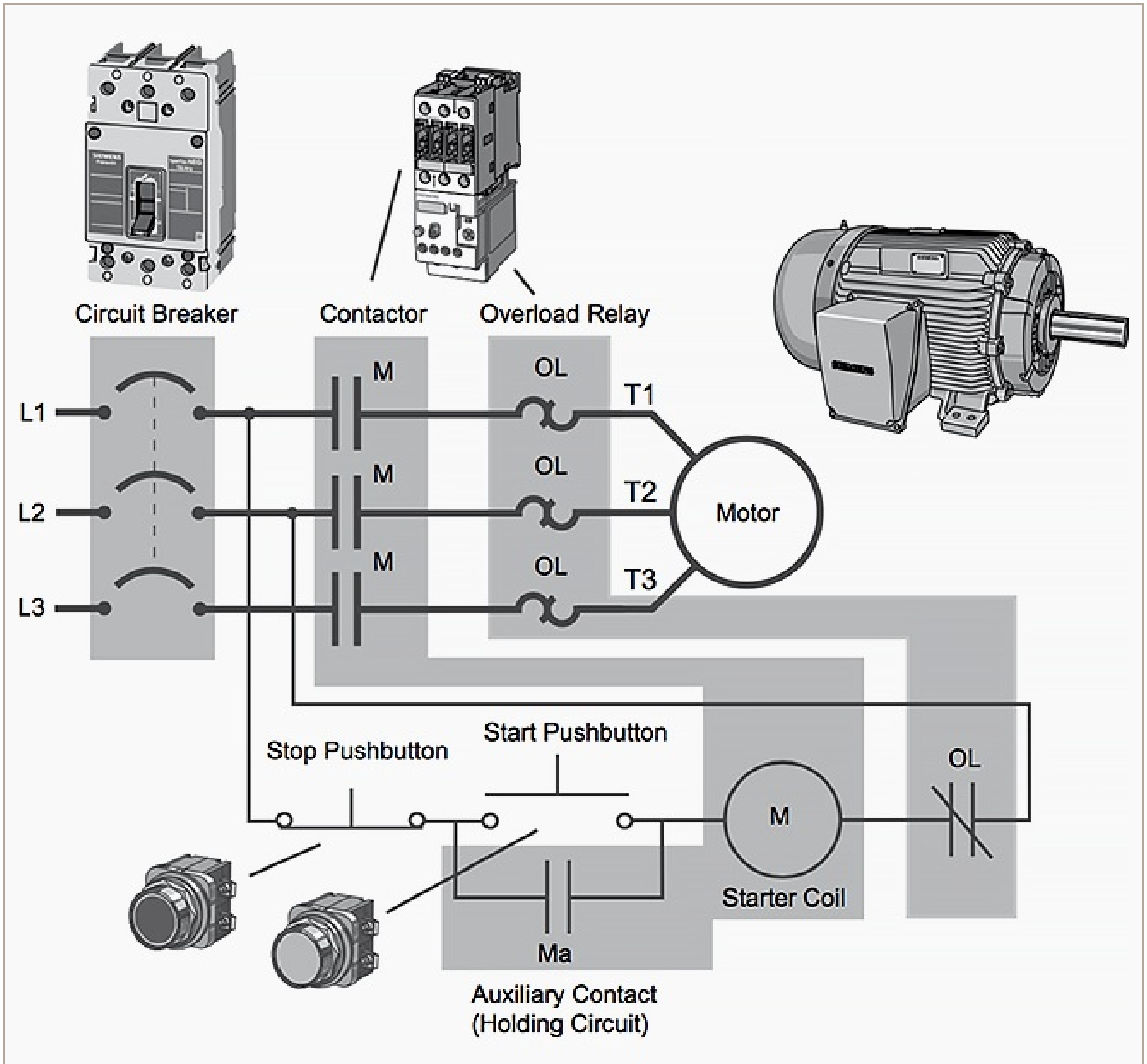
Basic PLC program for control of a three-phase AC motor - for beginners (on photo: Simatic S7-1500 PLC; credit: SIEMENS)

Motor Starter //

While the lighting control system previously discussed is useful to explain basic PLC operation, a more practical, and only slightly more complex, application is start-stop control of an AC motor. Before examining the PLC program, first consider a hard-wired approach.

> *The following line diagram illustrates how a normally open and a normally closed pushbutton might be connected **to control a three-phase AC motor**.*

In this example, a **motor starter coil (M)** is wired in series with a normally open, momentary Start pushbutton, a normally closed, momentary Stop pushbutton, and **normally closed overload relay (OL) contacts**.



Motor starter wiring diagram

Momentarily pressing the Start pushbutton completes the path for current flow and energizes the motor starter (M). This closes the associated **M** and **Ma** (auxiliary contact located in the motor starter) contacts.

When the Start button is released, current continues to flow through the Stop button and the **Ma contact**, and the **M coil** remains energized.

The motor will run until the normally closed Stop button is pressed, unless the **overload relay (OL)** contacts open. When the Stop button is pressed, the path for current flow is interrupted, opening the associated M and Ma contacts, and the motor stops.

This is how hard-wired motor starter functions. Now, let's discuss a little bit about PLC running the same //

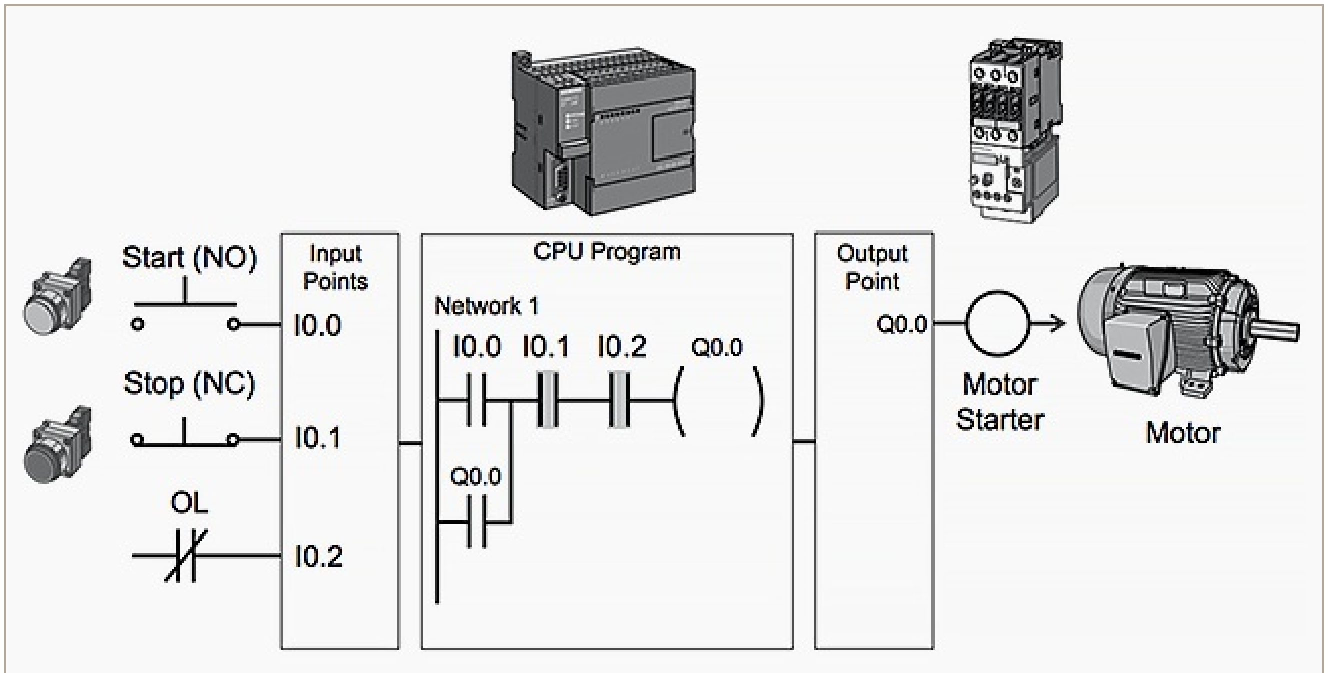
- PLC and motor control application
- PLC program operation
- Adding Run and Stop indicator lights to program

- Adding a limit switch to program
- Further expansion of a PLC program

PLC and motor control application

This motor control application can also be accomplished **with a PLC**. In the following example, a normally open Start pushbutton is wired to the first input (I0.0), a normally closed Stop pushbutton is wired to the second input (I0.1), and normally closed overload relay contacts (part of the motor starter) are connected to the third input (I0.2).

> *These inputs are used **to control normally open contacts in a line of ladder logic** programmed into the PLC.*



PLC Motor Control scheme

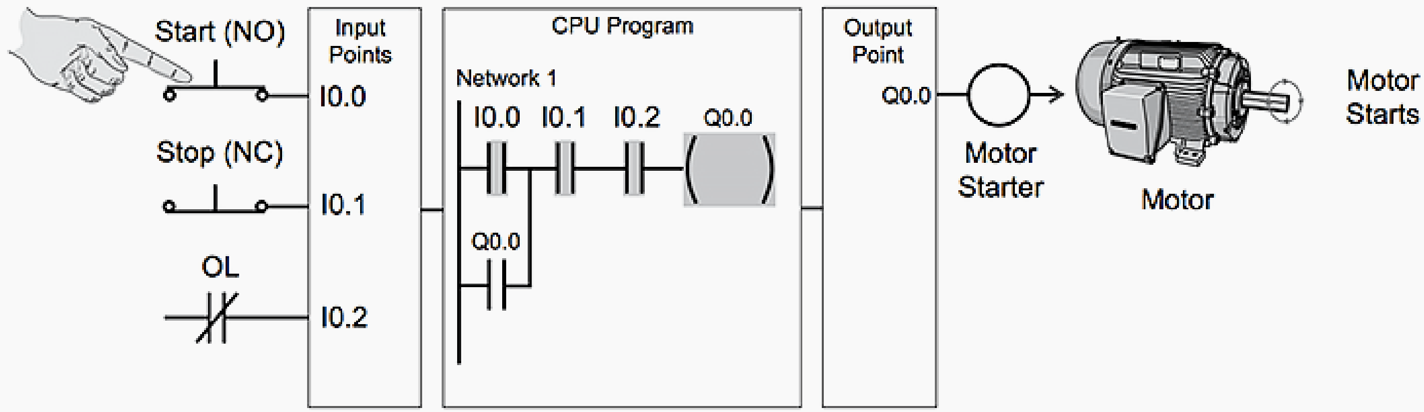
Initially, I0.1 status bit is a logic 1 because the normally closed (NC) Stop Pushbutton is closed. I0.2 status bit is a **logic 1** because the normally closed (NC) overload relay (OL) contacts are closed. I0.0 status bit is a **logic 0**, however, because the normally open Start pushbutton has not been pressed.

Normally open output Q0.0 contact is also programmed on Network as a sealing contact. With this simple network, energizing output coil Q0.0 is required to turn on the motor.

Go back to main topics ↑

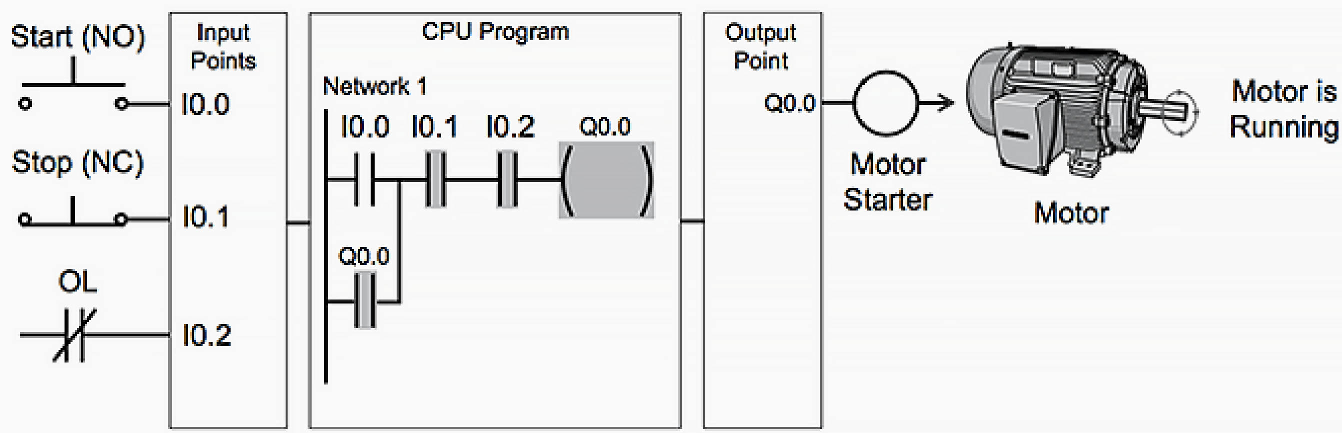
PLC program operation

When the Start pushbutton is pressed, the CPU receives a logic from input I0.0. This causes the I0.0 contact to close. All three inputs are now a logic . The CPU sends a logic to output Q0.0. The motor starter is energized and the motor starts.



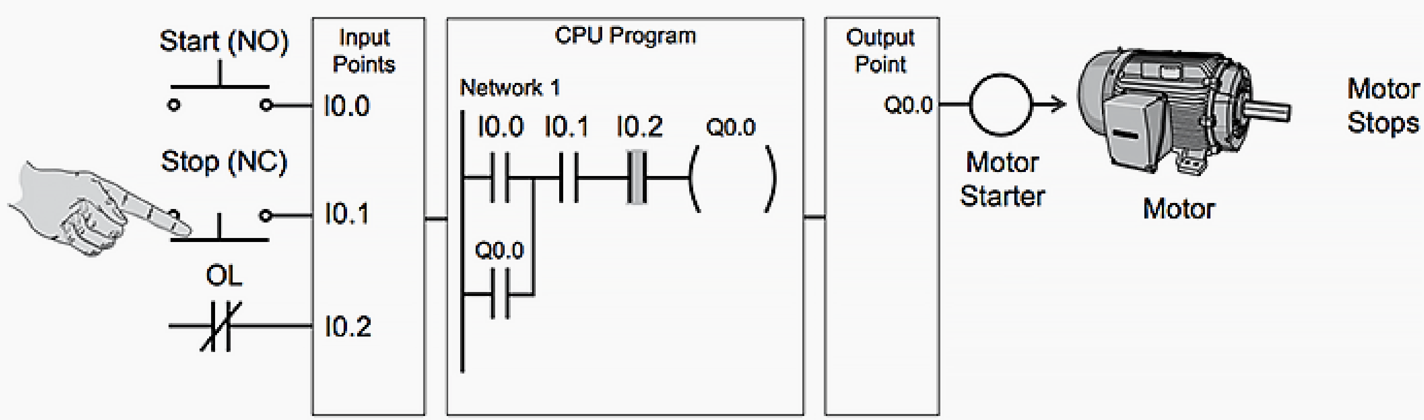
PLC Program Operation

The output status bit for Q0.0 is now a . On the next scan, when normally open contact Q0.0 is solved, the contact will close and output Q0.0 will stay on even if the Start pushbutton is released.



PLC Program Operation

When the Stop pushbutton is pressed: input I0.0 turns off, the I0.0 contact opens, output coil Q0.0 de-energizes, and the motor turns off.



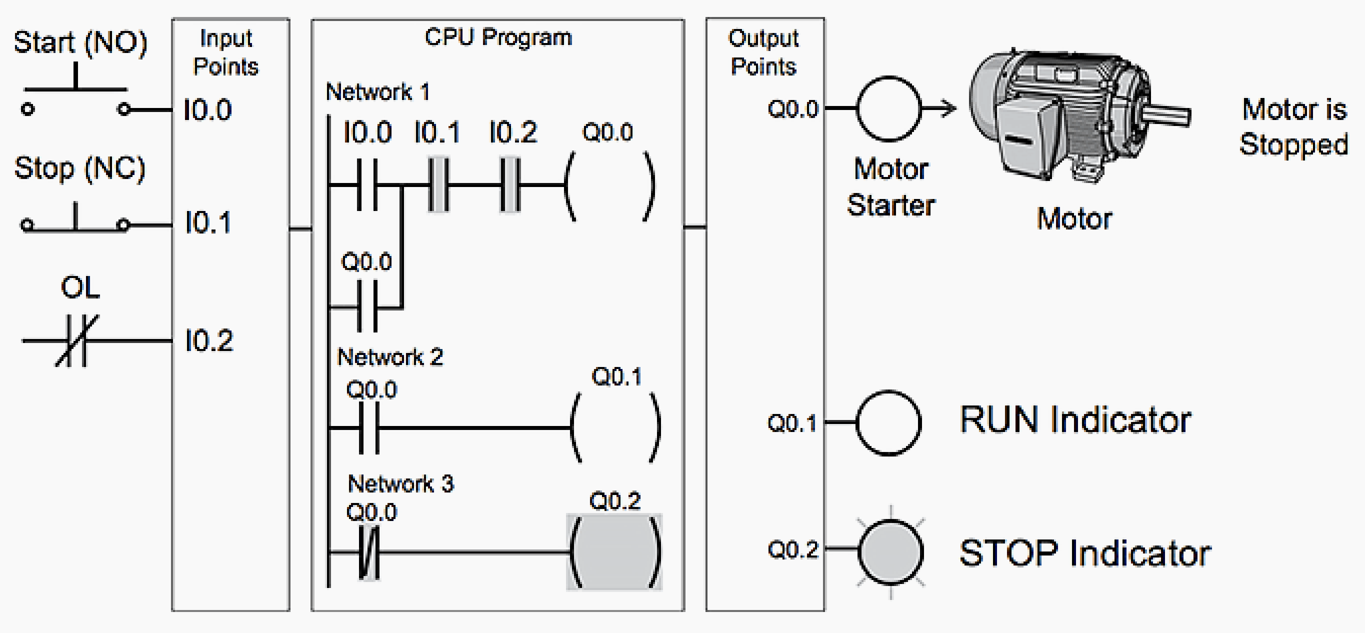
PLC Program Operation

Go back to main topics ↑

Adding Run and Stop indicator lights

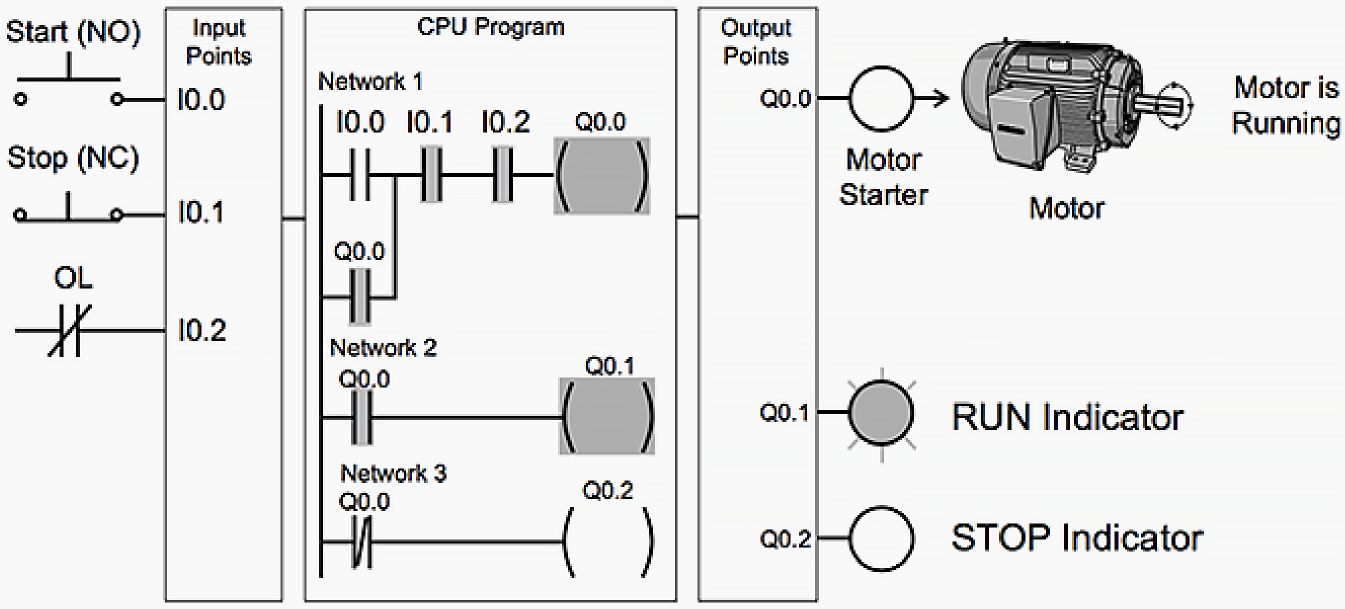
The application can be easily expanded **to include indicator lights for run and stop conditions**. In this example, a RUN indicator light is connected to output Q0. and a STOP indicator light is connected to output Q0.2.

The ladder logic for this application includes normally open Q0.0 contact connected on Network 2 to output coil Q0. and normally closed Q0.0 contact connected on Network 3 to output coil Q0.2. When Q0.0 is off, the normally open Q0.0 contact on Network 2 is open and the RUN indicator off. At the same time, the normally closed Q0.0 contact is closed and the STOP indicator is on.



Adding Run and Stop Indicator Lights

When the Start button is pressed, the PLC starts the motor. Output Q0.0 is now on. Normally open Q0.0 contact on Network 2 is now closed and the RUN indicator is on. At the same time, the normally closed Q0.0 contact on Network 3 is open and the STOP indicator light connected to output Q0.2 is off.



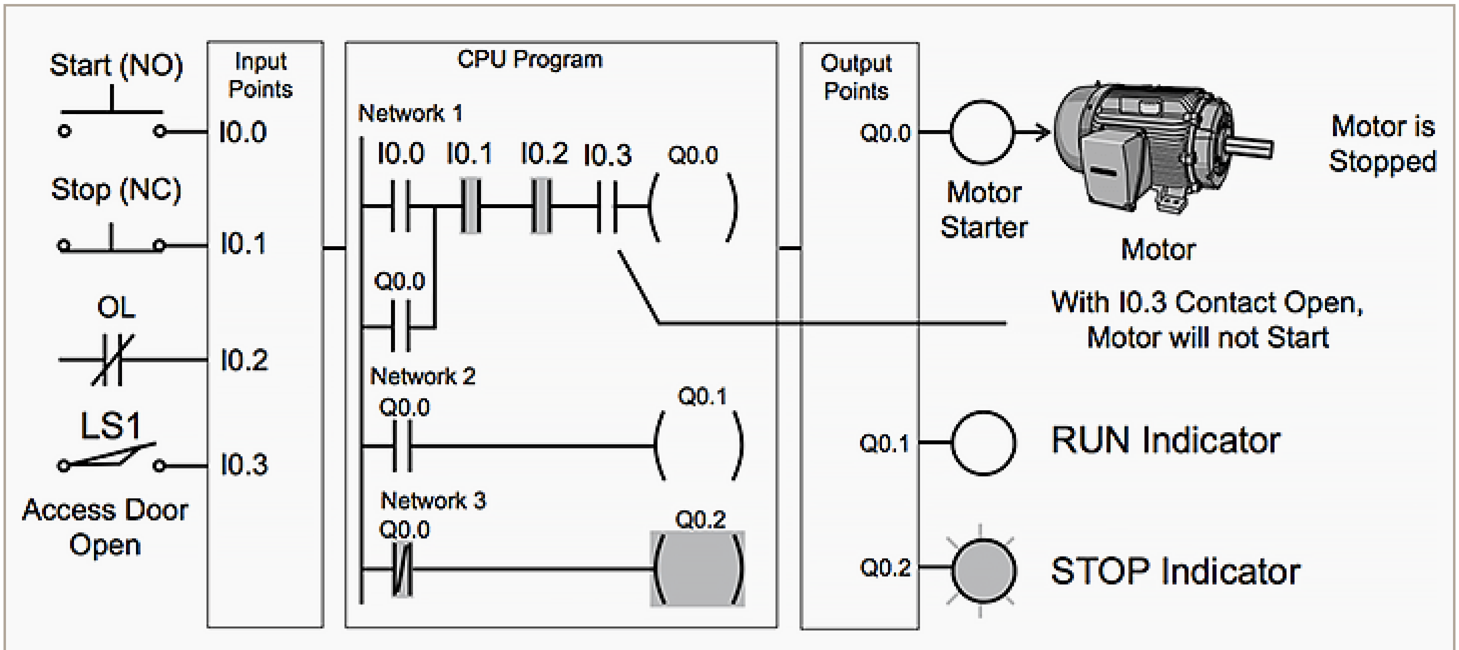
Adding Run and Stop Indicator Lights

Go back to main topics ↑

Adding a limit switch

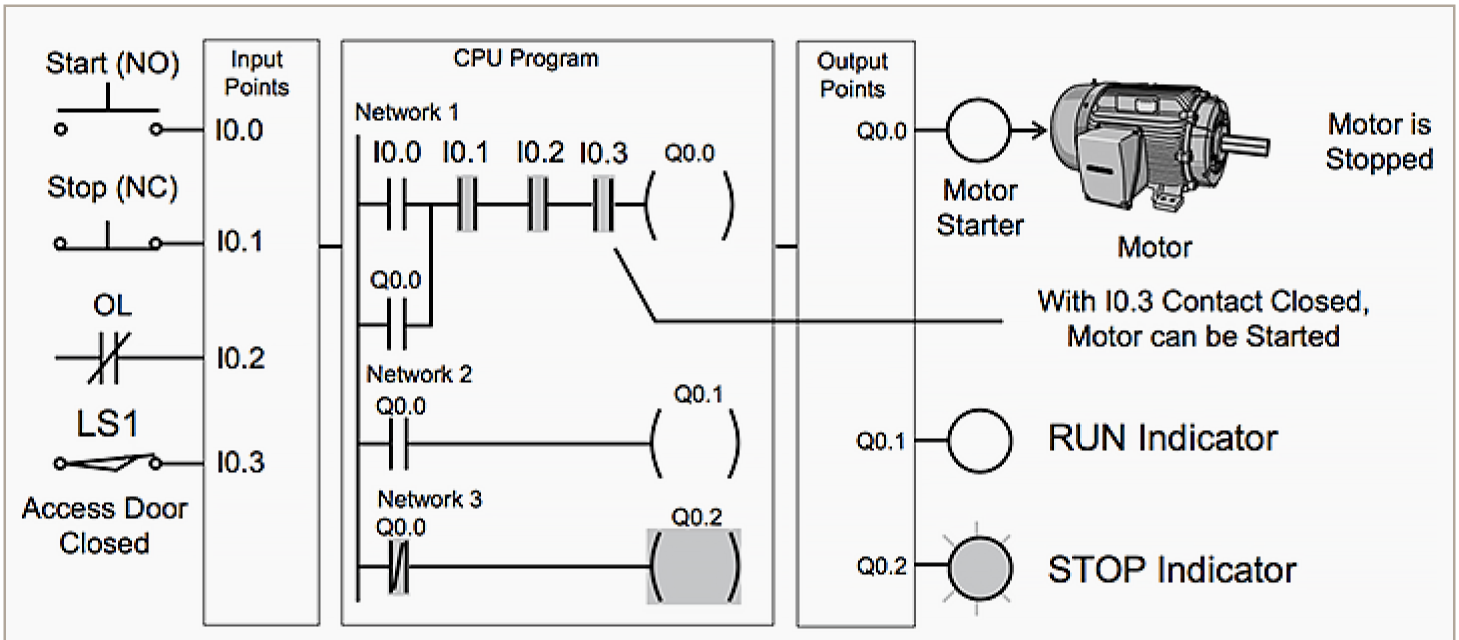
The application can be further expanded by adding a limit switch. The limit switch could be used in this application for a variety of functions. For example, the limit switch could be used to stop the motor or prevent the motor from being started.

In this example, the limit switch is associated **with an access door to the motor** or its associated equipment. The limit switch is connected to input I0.3 and controls a normally open contact in the program. If the access door is open, limit switch LS is open and normally open contact I0.3 is also open. **This prevents the motor from starting.**



Adding a Limit Switch

When the access door is closed, **limit switch LS is closed** and normally open contact I0.3 is also closed. This allows the motor to start when the Start pushbutton is pressed.



Adding a Limit Switch

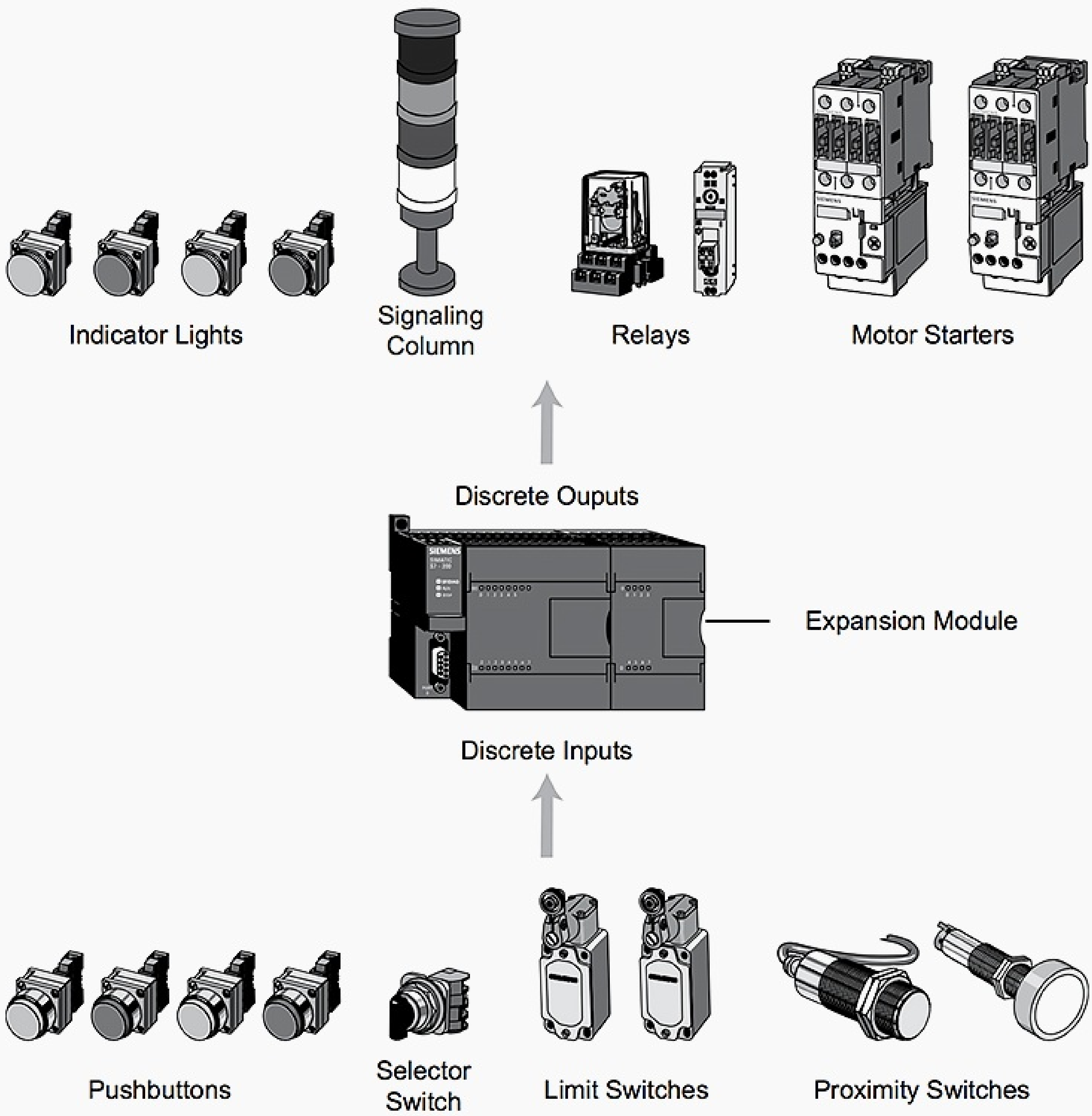
Go back to main topics ↑

Further expansion of a PLC program

The PLC program can be further expanded to accommodate a wide variety of commercial and industrial applications.

Start/Stop pushbuttons, selector switches, indicator lights, and signaling columns can be added. Motor starters can be added for control of additional motors. Over-travel limit switches can be added along with proximity switches for sensing object position. Various types of relays can be added to expand the variety of devices being controlled.

As needed, expansion modules can be added **to further increase the I/O capability**. The applications are only limited by the number of I/Os and amount of memory available for the PLC.



Further expansion of a PLC program

Go back to main topics ↑

***Reference** // The basics of PLCs by SIEMENS*