Functional Blocks: Analog Inputs and Flow Control

If an input pin is 0V or +24V, it is read as a binary signal

- 0V = false
- +24V = true

If an input pin is in the range of [0V , 10V], it is read as an analog input with a value of 0 .. 1000.

Using comparitors, you can turn on and off relays based upon the input voltage.

## Example 1: Turn on the lights at 2.00V / 4.00V / 6.00V

Name	Data Type	Dimension	Alias	Initial Value	Attribute	
	WORD					
W200	WORD			200	Read/Write	
W400	WORD			400	Read/Write	
W600	WORD			600	Read/Write	

Tabs: User Global Variables - Micro810 | Local Variables - UntitledLD | System Variables - Micro810 | I/O - Micro810 | Defined Words - Micro810

Example 2: On-Off Control - Hysteresis

Control the voltage of an RC filter to be in the range of 4 - 6V.
- Turn on at 4.00V
- Turn off at 6.00V
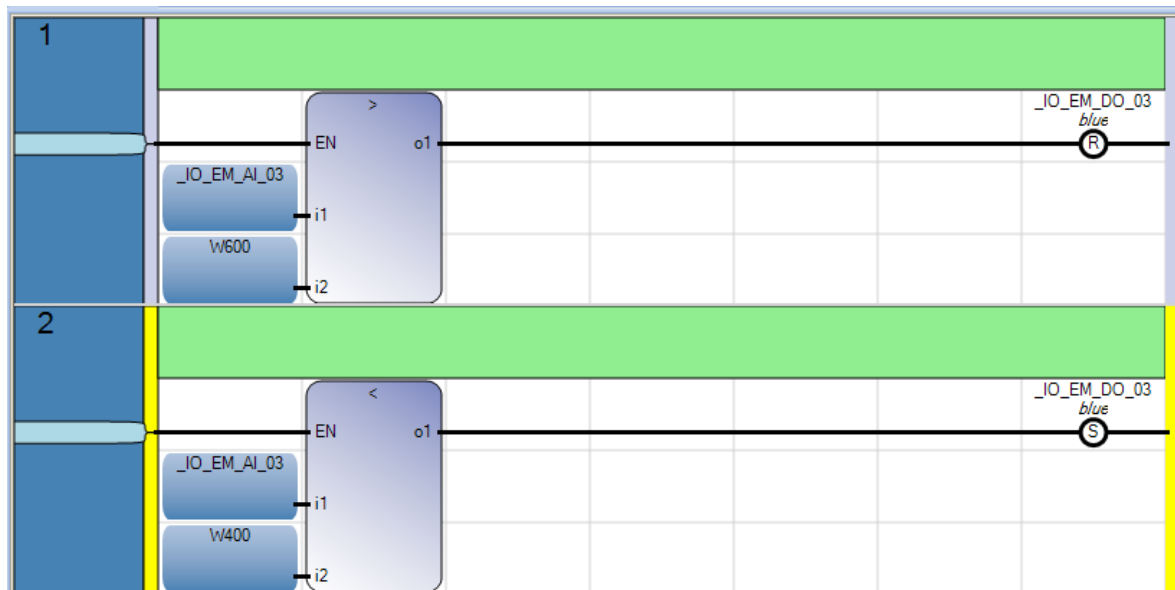
Flow Control:

Have three modes of operation:
- High: Regulate the voltage to be in the range of (6V, 8V)
- Medium: Regulate the voltage to be in the range of (4V, 6V)
- Low: Regulate the voltage to be in the range of (2V, 4V)

Inputs:
- DI_00: Set to High mode
- DI_01: Set to Medium mode
- DI_02: Set to Low mode

Outputs:
- DO_00: (red) Indicate that you are in high mode
- DO_01: (yellow) Indicate that you are in medium mode
- DO_02: (green) Indicate that you are in low mode

- DO_03 On-Off for a motor / pump / power supply. When on, charges up the voltage on AI_03 to +10.00 volts through an RC filter (first-order differential equation).

Note that ladder logic is a program: it starts executing on rung 1 and works its way down. When it reaches the end of the program, it stops and starts executing again in 10ms. There is a *jump* command which allows you to skip over part of a program.

The following program works as follows:
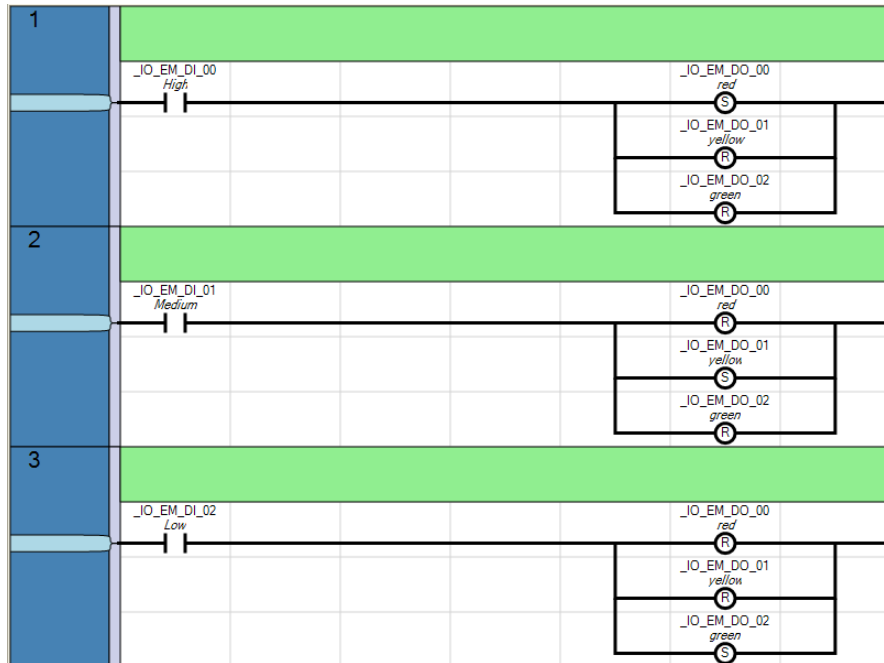
Rung 1..3: Check the digital input buttons DI_00 to DI_02 to set and clear the indicator LEDs (red / yellow / green). Normally these would be internal states. Relay outputs are used here to make it easier to see what is going on.
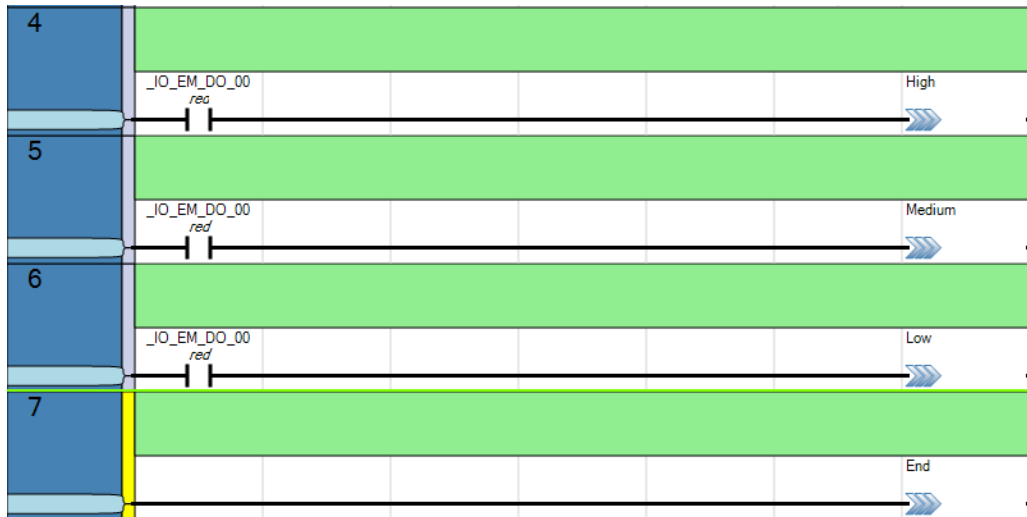
Rung 4..7: Flow control. Jump to various sections of the program based upon what mode you are in. If you are not in any mode, keep checking the input buttons until you press a button.
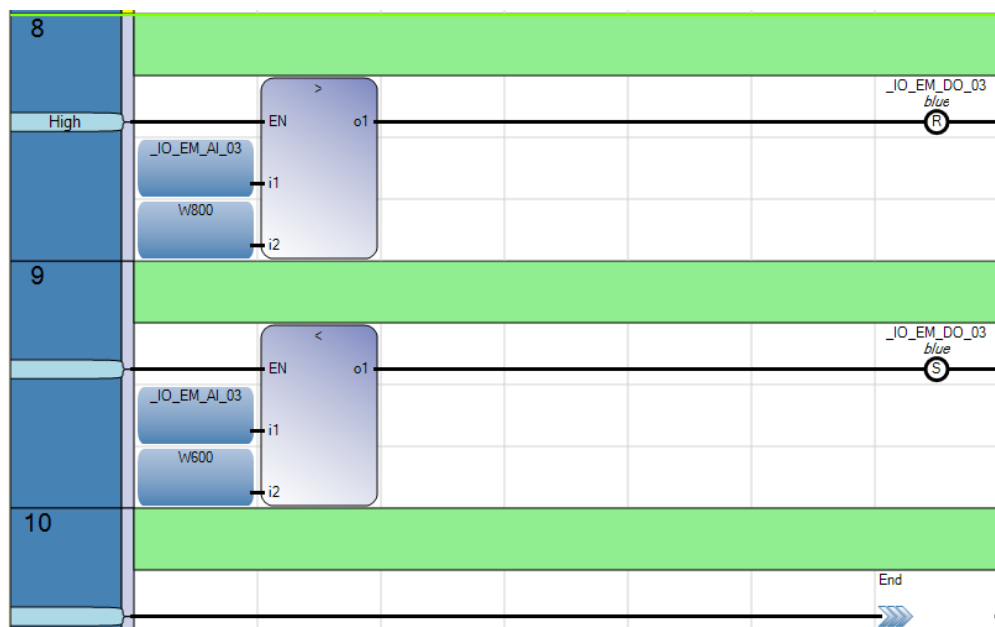
Rung 8-10: High-Level Control. Close the relay when V < 6V and off when V > 8V.
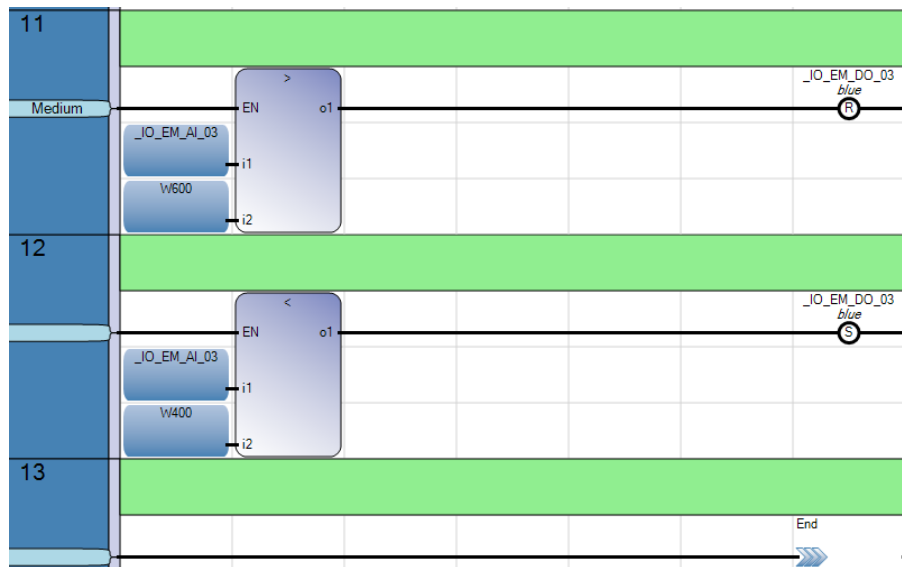
Rung 11-13: Medium Level Control. Close the relay when V < 4V and off when V > 8V
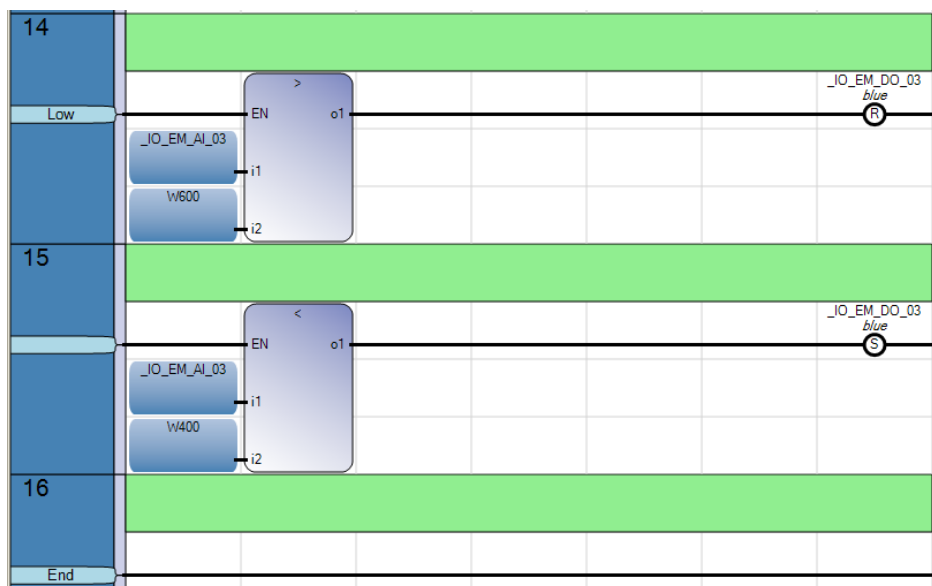
Rung 14-15:   Low-Level Control. Close the relay when V < 2V and off when V > 4V

Rung 16: The end of the program. Note that the program executes starting on rung #1 every 10ms. The program needs to end for it to restart again in 10ms.

1

_IO_EM_DI_00
High
─┤ ├─

_IO_EM_DO_00
red
(S)

_IO_EM_DO_01
yellow
(R)

_IO_EM_DO_02
green
(R)

2

_IO_EM_DI_01
Medium
─┤ ├─

_IO_EM_DO_00
red
(R)

_IO_EM_DO_01
yellow
(S)

_IO_EM_DO_02
green
(R)

3

_IO_EM_DI_02
Low
─┤ ├─

_IO_EM_DO_00
red
(R)

_IO_EM_DO_01
yellow
(R)

_IO_EM_DO_02
green
(S)

4

_IO_EM_DO_00
red
─┤ ├─

High
»

5

_IO_EM_DO_00
red
─┤ ├─

Medium
»

6

_IO_EM_DO_00
red
─┤ ├─

Low
»

7

End
»

11

Medium

>

EN   o1

_IO_EM_AI_03

i1

W600

i2

_IO_EM_DO_03
blue
(R)

12

<

EN   o1

_IO_EM_AI_03

i1

W400

i2

_IO_EM_DO_03
blue
(S)

13

End

11

Medium

>

EN   o1

_IO_EM_AI_03

i1

W600

i2

_IO_EM_DO_03
blue
(R)

12

<

EN   o1

_IO_EM_AI_03

i1

W400

i2

_IO_EM_DO_03
blue
(S)

13

End

14

Low

>

EN   o1

_IO_EM_AI_03

i1

W600

i2

_IO_EM_DO_03
blue
(R)

15

<

EN   o1

_IO_EM_AI_03

i1

W400

i2

_IO_EM_DO_03
blue
(S)

16

End