Chapter 2: Basic Ladder Logic Programming
Ladder Logic

Learning objectives

- Understand basic ladder logic symbol
- Write ladder logic for simple applications
- Translate relay ladder logic into PLC ladder logic
Simple Ladder Logic

Ladder Logic:

- Primary Programming Language for PLCs.
- Visual and Graphical language unlike textual high-level, such as C, C++, Java…
- Derived from relay logic diagrams
- Primitive Logic Operations:
  - OR
  - AND
  - NOT
Simple Ladder Logic

OR Operation

- Control Behavior: The light should be on when either switch A is on (i.e., closed) or switch B is on (closed). Otherwise it should be off.

- Task: Implement this behavior using
  - Relay circuit
  - PLC Ladder Logic
Simple Ladder Logic

**OR Operation**

- Possible Combinations of the 2 Switches: \(2^2\)

**OR Truth Table**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Light</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>ON</td>
</tr>
</tbody>
</table>
**OR Operation

Relay Circuit**

- Switches A and B are connected in parallel to relay coils AR & BR resp.
- When switch A (or switch B) is closed relay coil AR (or BR) gets energized
  - The *Normally Open (NO)* contact AR (or BR) gets closed
    - Power is transmitted to coil LR
      - Relay coil LR gets energized
        - The *NO* contact LR gets closed
        - **Power is transmitted to the Light bulb**
OR Operation
Relay Ladder Logic Circuit

120v

Neutral

A
B
AR
BR
LR
W
BR

Lamp
OR Operation

PLC Ladder Logic

- Append above to the leading two rungs of relay ladder logic diagram.
- Switch A and Switch B are connected to discrete input channels of the PLC.
- Light is connected to discrete output channel (actuator) of the PLC.

When input switch A (or switch B) is on, the light is on.
Simple Ladder Logic

AND Operation

- Control Behavior: The light should be on when switch A is on (i.e., closed) and switch B is on (closed). Otherwise it should be off.

- Task: Implement this behavior using
  - Relay circuit
  - PLC Ladder Logic
Simple Ladder Logic

**AND Operation**

- Possible Combinations of the 2 Switches: \(2^2\)

### AND Truth Table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Light</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>ON</td>
</tr>
</tbody>
</table>
Switches A and B are connected in series to relay coils AR & BR respectively.

When switch A is closed relay coil AR gets energized

- The *Normally Open (NO)* contact AR gets closed
  - Power flows to *Normally Open (NO)* contact BR, where it terminates until BR is energized
    - Subsequently, when BR gets energized, LR is energized, which causes the *NO* contact LR to close
      - Power is transmitted to the Light bulb

What happens if BR is energized before AR?
AND Operation

PLC Ladder Logic Circuit

\[ V^+ \quad \text{A} \quad \text{B} \quad \text{Light} \quad \text{Com} \]
Simple Ladder Logic

NOT Operation

- **Control Behavior:** The light comes on only when switch A is on (i.e., closed) and switch B is off (open). Otherwise it should be off.

- **Task:** Implement this behavior using
  - Relay circuit
  - PLC Ladder Logic
Simple Ladder Logic

**NOT Operation**

- Possible Combinations of the 2 Switches: \(2^2\)

### NOT Truth Table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Light</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
</tr>
</tbody>
</table>
### NOT Operation

#### Relay Circuit

- Switches A and B are connected to relay coils AR & BR resp.
- When switch A is closed relay coil AR gets energized
- When switch B is off (on) relay coil BR is not energized (energized) and BR contact is normally-closed (normally-open)
  
  ![Circuit Diagram]

  ```
  V+                   
  |                   |
  |                   |
  A  AR              L1
  |     |              |
  |     |              |
  Com --> B  BR      --> Light
  |     |              |
  |     |              |
  L1  LR
  ```

……
NOT Operation
PLC Ladder Logic

\[ V^+ \]
\[ A \quad B \quad \text{Light} \quad \text{Com} \]
Simple Ladder Logic

NAND Operation

NAND (NOT AND)

- Control Behavior: The light comes on only when switch A is off and switch B is off. Otherwise it should be off.

- Task: Implement this behavior using
  - Relay circuit
  - PLC Ladder Logic
Simple Ladder Logic

NAND Operation

- Possible Combinations of the 2 Switches: \((2^2)\)

### NAND Truth Table

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Light</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
</tr>
</tbody>
</table>
NAND Operation
Relay Circuit
NAND Operation
Ladder Logic Circuit
Digital Logic

**Gates**

**AND**

\[ A \cdot B \]

**OR**

\[ A + B \]

**NAND**

\[ \overline{A \cdot B} \]

**NOR**

\[ \overline{A + B} \]

**NOT**

\[ \overline{A} \]
Basic Ladder Logic Symbol

**Normally open contact**
Passes power (ON) if coil driving the contact is ON (closed)
Allen-Bradley calls it **XIC - eXamine If Closed**

**Normally closed contact**
Passes power (ON) if coil driving the contact is **off** (open)
Allen-Bradley calls it **XIO - eXamine If Open**

**Output or coil**
If any left-to-right path of inputs passes power, output is energized
Allen-Bradley calls it **OTE - OuTput Energize**

**Not Output or coil**
If any left-to-right path of inputs passes power, output is de-energized

The IEC 61131-3 standards describe the complete list of ladder logic contact and coil symbols. **See also section 2.3.1**
PLC Ladder Logic Symbols

- The symbols are ladder logic instructions
- The PLC scans (executes) the symbols:

  \[
  \begin{align*}
  \rightarrow &\quad = \text{on} = \text{Closed} = \text{True} = 1 \\
  \rightarrow/ &\quad = \text{off} = \text{Open} = \text{False} = 0
  \end{align*}
  \]

- Every PLC manufacturer uses instruction symbols
- Industry trend is based on IEC 61131-3
  - Variations in symbols by Manufacturers
- Allen-Bradley ControlLogix symbols slightly different (Refer 2.3.3)
Ladder Logic Diagram

- Power Rails - Pair of Vertical Lines
- Rungs - Horizontal Lines
- Contacts A, B, C, D… arranged on rungs
- Note in PLC Ladder Logic:
  - No Real Power Flow (like in relay ladder)
  - There must be continuous path thru’ the contacts to energize the output
Two Classes of Ladder Logic Instructions

- **Output:** Appears on extreme RHS of rung always – Out1, Out2
- **Input:** Any instruction that can replace a contact

Can contacts appear on the RHS of a coil?
Ladder Logic Diagram

Function Block Instructions

- Any non-contact instruction:
  - Timer Instruction
  - Counter Instruction
  - Comparison Instruction
Ladder Logic Diagram
Example 1

**Task:**

Draw a ladder diagram that will cause the output, pilot light PL2, to be on when selector switch SS2 is closed, push button PB4 is closed and limit switch LS3 is open. (Note: no I/O addresses yet.)

**Thought Process**

- Identify the output: PL2 → PL2 appears on rhs of rung
- What is the behavior (type of connection to use):
  - sequential operation of all switches → series connection
- Type of contacts to implement output:

```
SS2 closed ┐ │ ┤ PB4 closed ┐ │ ┤ LS3 open ┐ / ┤
```

Ladder Logic Diagram

Example 1

SS2  PB4  LS3  PL2
Ladder Logic Diagram
Example 2

Task:

Draw a ladder diagram that is equivalent to the following digital logic diagram

Y is on when (A is on, B is on and C is off) or D is on, or E is off

What is the Boolean logic expression?
Ladder Logic Diagram

Example 2

- Thought Process
  - Identify the output: Y → Coil Y appears on rhs of rung
  - What is the behavior (type of connection to use):
    - The inputs A, B, C for AND gate will be connected in series
    - The D, E inputs for OR gate will be connected in parallel with the output of AND gate
  - Type of contacts to implement output (review the expected behavior again to determine contact types):

    A is on:    B is on:    C is off:  → / →
    |   |       |   |       |   |       

    D is on:    E is off:  → / →
    |   |       |   |       

What happens if the D contact refers to Y?
Ladder Logic Diagram

Sealing an output

Output Y is set (latched) indefinitely
Ladder Logic Diagram Dangers

Repeated Output

- Do not repeat normal output coils that refer to the same address

- The coils for first and second rung refer to Out1

- Second rung overrides the logic in first rung
First consider the output

- Next, consider ALL the conditions that drive the output (Out1) (Implement the conditions in parallel)
Ladder Logic Diagram Dangers

- Use set/seal (latch) and reset (unlatch) together:
  - If a set coil refers to an output there should be a reset coil for that output
  - Reverse power flow in contact matrix is not allowed
    - Power flow one way left to right (solid state relays)
Ladder Logic Diagram Dangers
Reverse Power Flow

- This is not allowed:

- If the reverse power flow path is truly needed, then put it as a separate path, where the power flows from left to right:
Typical PLC Processor Scan

- **Major tasks in a scan**
  - Read Inputs
  - Execute Ladder Logic
  - Update Outputs
  - Scan Time

- Processor must read the state of the physical inputs and set the state of the Physical outputs
Typical PLC Processor Scan

- **Order of PLC Processor Scan**
  - Read Physical Inputs
  - Scan ladder logic program
  - Write the physical outputs

- **Scan Time**
  - Time to complete above cycle
  - Order of 1-200 milliseconds

What could happen if scan time exceeds more than 200 milliseconds?
The state of actual input devices are copied to an area of the PLC Memory, **input data table** before the ladder logic program executes.

- As the ladder logic program is scanned, it reads the input data table then writes to a portion of PLC memory - **the output data** table as it executes.

- The output data table is copied to the actual output devices **after** the ladder logic has been scanned.

**What is the significance of the input and output data tables?**
The execution of PLC Processor controlled by processor mode (Refer to lab 1)

- **Run Mode:**
  - Physical Input, Physical Outputs and Ladder logic all get scanned

- **Remote Mode**
  - Download ladder logic to PLC Processor; and initiate scan from the remote terminal

- **Program Mode**
  - Ladder logic not scanned
Ladder Logic Evaluation

- For most PLC's, the ladder scan starts at the top of the ladder and proceeds to the bottom of the ladder, examining each rung from left to right.
  - Once a rung is examined, it is not examined again until the next scan.
  - The rungs are not examined in reverse order.
- The JMP instruction may be used to jump back up the ladder and execute earlier rungs.
  - Use of JMP not recommended  Why?
Ladder Logic Evaluation
Push Button (PB)

Start PB:

Physical Input: PB
Physical Outputs: PL1, PL2, PL3 and PL4

Start of PLC scans
State of PLC image memory for I/O devices:
Ladder Logic Evaluation

Push Button (PB)

Scan 1: Only the state of PB changes to ON (1) during the scan.

Scan 2: The ON state of PB is copied into Input data table before Ladder logic is scanned.

When rung 1 is scanned → PL1 is still off (0)
When rung 2 is scanned → PL2 is still off (0) Why?
What is the value of PL4 and PL3 in Output Data table?

When rung 3 is scanned the Value of PL3 in the output data table changes to 1 Why?

When rung 4 is scanned, the Value of PL4 in the output data table remains at off (0). Why?

At the end of scan 2 the values in Output data table are copied to the Physical Output Devices. PL 3 turns on.
Scan 3:
When rung 1 is scanned the value of PL4 in output data table is still 0 → PL1 in output data table remains 0
When rung 2 is scanned the value of PL3 in Output Data table is currently 1 → value of PL2 in Output Data table changes to 1

When rung 3 is scanned the Value of PB in the input data table is still 1 → Value of PL3 in Output data table remains at 1

When rung 4 is scanned Value of PL2 in the output data table is now 1 so the value of PL4 in the Output Data table changes to 1

At the end of scan 3 the values in Output data table are copied to the Physical Output Devices:
PL2 and PL4 turn on simultaneously
(PL3 remains on)
Scan 4:
When rung 1 value of PL4 in output data table is now 1 → value of PL1 in output data table changes to 1
When rung 2 is scanned the value of PL3 in Output Data table is still 1 → value of PL2 in Output Data table remains at 1

When rung 3 is scanned the Value of PB in the input data table is still 1 → Value of PL3 in Output data table remains at 1

When rung 4 is scanned Value of PL2 in the output data table is still 1 so the value of PL4 in the Output Data table remains at 1

At the end of scan 4 the values in Output data table are copied to the Physical Output Devices:
**PL1 turns on**
(PL2, PL3 and PL4 remain on)
Ladder Logic Evaluation

Push Button (PB)

**Scans 5 and 6:** Nothing Changes

Scans 7 – 9: Similar to Scans 2 – 4 except that state changes from 1 (on) to 0 (off)
Ladder Logic Evaluation

Push Button (PB) Scan Timing Diagram

I/O Terminal: -------
I/O Data Table: _____
**Ladder Logic Evaluation**

**Push Button (PB1)**

Assume rungs are scanned from top - down

Physical Input: PB1

Physical Output: PL1
Ladder Logic Evaluation
Push Button (PB1)

Scan 1: Only the state of PB1 changes to **ON (1)**
  during the scan, new state copied at next scan

Scan 2:
The **ON** state of PB1 is copied into Input data table
  before Ladder logic is scanned

When rung 1 is scanned, PB1 is ON, Int3 is off so power
  goes to Int2, But Int2 is off → Int1 is off (0)
When rung 2 is scanned PB1 is ON, Power goes thru’ Int3
  and Int1 → Int2 is On
When rung 3 is scanned the Value of PB1 in the input data
  table is now 1 so Int3 is energized and Int3 contact is ON
When rung 4 is scanned, the Value of Int2 in the PLC memory
  is now 1 so the value of PL1 in the Output data table changes to 1

At the end of scan 2 the values in Output data table are copied
to the Physical Output Devices. **PL1 turns on**
Ladder Logic Evaluation

Push Button (PB1)

Scan 3: No change in the rung output coils
When rung 1 is scanned There is continuity thru’ PB1 and Int2 but not Int3

When rung 2 is scanned – no continuity thru’ top branch
But continuity thru’ lower branch → Int2 remains ON

At the end of scan 3 the values in Output data table are copied to the Physical Output Devices. PL1 remains on
Ladder Logic Evaluation

Push Button (PB1)

Scans 4 - 5:

No change in the rung output coils because there is no change in the contacts

At the end of scans 4 and 5 the values in Output data table are copied to the Physical Output Devices. PL1 remains on
Discrete Input/Output

- An actual PLC has connections to the "real" world, and is not just ladder logic.

An example hard-wired ladder circuit
Programming with NC Contacts

- If you want “action” (turn ON) when switch is **closed** (relay energized), use .
- If you want “action” (turn ON) when switch is **open** (relay de-energized), use .

In the rungs, think of the contact as a symbol,

<table>
<thead>
<tr>
<th></th>
<th>= ON = CLOSED = TRUE = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>= OFF = OPEN = FALSE = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: this is probably the most confusing concept in ladder logic