Topics

- PLC programming languages
- Anatomy of a ladder program
- Logic functions
- Logical continuity vs. electrical continuity
- I/O Mapping
- Mastering examine on and examine off instructions
- The PLC scanning process
PLC Programming Languages

- In the United States, **ladder logic** is the most popular method used to program a PLC.
- This course will focus primarily on ladder logic programming.
- Other programming methods include:
  - Function block diagrams (FBDs)
  - Structured text (ST)
  - Instruction List (IL)
  - Sequential function charts (SFCs)

Anatomy of a Ladder Program

Input Instructions (conditions) | Output Instructions (actions)
--- | ---
Rung 0 |  
Rung 1 |  
Rung 2 |  
Left power rail | Right power rail
Anatomy of a Ladder Program (cont’d)

- Input instructions are entered on the left
- Output instructions are entered on the right
- The power rails simulate the power supply lines
  - L1 and L2 for AC circuits and +24 v and ground for DC circuits
- Most PLCs allow more than one output per rung

Anatomy of a Ladder Program (cont’d)

- The processor (or “controller”) scans ladder rungs from top-to-bottom and from left-to-right.
  - The basic sequence is altered whenever jump or subroutine instructions are executed.
Anatomy of a Ladder Program (cont'd)

A 3-rung example ladder program

Control the escalator motor and the green stack light

Rung Comment

Rung Number

Description assigned to alias tag
Alias tag pointing to base address
Base address
Input Instruction
Logic Functions

- PLC programming is a logical procedure.

- In a PLC program, “things” (inputs and rungs) are either TRUE or FALSE.

- If the proper input conditions are TRUE:
  - The rung becomes TRUE and an output action occurs (for example, a motor turns on).

- If the proper input conditions are not TRUE:
  - The rung becomes FALSE and an output action does not occur.

Logic Functions (cont’d)

- Ladder logic is based on the following logic functions:
  - **AND**
  - **OR**
    - Sometimes called “inclusive OR”
  - **Exclusive OR**
Logic Functions - AND

<table>
<thead>
<tr>
<th>Input 1</th>
<th>Input 2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

0 → False
1 → True

Contacts ANDed together

Logic Functions - OR

<table>
<thead>
<tr>
<th>Input 1</th>
<th>Input 2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

0 → False
1 → True

Contacts ORed together
Logic Functions - Exclusive OR

In addition to ANDing and ORing, the Exclusive OR (XOR) is also useful. When the inputs are DIFFERENT, the XOR output is true.

<table>
<thead>
<tr>
<th>Input 1</th>
<th>Input 2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Output: 0 → False
          1 → True

Logic Functions (cont’d)

Example 1 – Inputs ANDed together in series

Inputs: B_System_On, Photoeye 3, Photoeye 4

Output: Product Stop 2

Three input instructions ANDed together.

All 3 input instructions must be true in order to energize the output
Logic Functions (cont’d)

Example 2 – Inputs ORed together in parallel

Three input instructions ORed together.

If any of the 3 input instructions are true the output will be energized.

Logic Functions (cont’d)

Example 3 – A combination of ANDing and ORing
Logical Continuity

- **Logical continuity** in a ladder rung occurs when there is a continuous path of TRUE conditions from the left power rail to the output instruction(s).

- When there is logical continuity, the rung becomes true and the output becomes energized.

### Logical Continuity – Example 1

#### Rung 0

- **T**
- **F**
- **T**

Rung False

This input instruction is false
This input instruction is true

#### Rung 1

- **F**

Rung False

#### Rung 2

- **F**
- **T**
- **T**

Rung True

Left power rail
Right power rail
Logical Continuity – Example 2

Virtual power flow (not actual current flow)

I010/0 → I010/5 → O010/7

Path of logical continuity

I010/1 → R3/10

I010/6 → O010/4 → O010/5

TRUE conditions are highlighted in **green** on the programming panels display.

---

Logical Continuity vs. Electrical Continuity

- **Electrical continuity** in an input circuit, occurs when there is a complete path for current to flow.

- A PLC input circuit is a simple series circuit consisting of a:
  - Power supply,
  - Switch, and a
  - Load

- When there is electrical continuity, a bit in the PLCs memory (sometimes called the input image table) is set to a 1.
Electrical Continuity

Normally open (N.O.) pushbutton [shown pushed]

+24v

Input Module

Input Circuitry

COM

Power supply is implied

Screw terminals

Power rail

Power rail
Logical vs. Electrical Continuity

- Note: It’s possible to have electrical continuity and not have logical continuity (and vise versa).

Control Wiring Diagram:

PB0 - N.O. pushbutton (shown pushed)

Ladder Program:

This instruction is false when the pushbutton is pushed

Electrical Continuity

I/O Mapping

- Every discrete input is assigned to a specific bit in the PLC’s memory (input image table)
  - If there is electrical continuity, the bit is set to a 1
  - If there is no electrical continuity, the bit is reset to a 0

- Every discrete output is assigned to a specific bit in the PLC’s memory (output image table)
  - In order for an output to turn on, its associated bit must first be set to a 1
1 word (16 bits)

Input Image Table

Output Image Table

Note: Bit addresses are given in octal for an Allen-Bradley PLC-5 system.
The bits are numbered 0 – 17.

In RSLogix 5 software:

- Input Image Table
- Output Image Table

I/O Mapping
I/O Mapping

Output Image Table in RSLogix 5:

O:005 is a 16 bit word address:

O:005/07 is the address of a bit which resides in the word O:005

Input Image Table in RSLogix 5:

1 word = 16 bits (bits are numbered in octal for a PLC-5)
I/O Mapping

ControlLogix tag database:

Mastering Examine On & Examine Off Instructions

- Discrete input devices have normally open (N.O.) and/or normally closed (N.C.) contacts.

  - Example: Pushbuttons can be purchased with either N.O. or N.C. mechanical contacts.

  - “Normally” implies the state of the contacts when you are NOT pushing the button.
**Mastering Examine On & Examine Off Instructions**

- Normally open (N.O.) vs. normally closed (N.C.) contacts:

<table>
<thead>
<tr>
<th>Contact Type</th>
<th>Resistance between contacts when NOT pushed</th>
<th>Resistance between contacts when pushed</th>
</tr>
</thead>
<tbody>
<tr>
<td>N.O.</td>
<td>Infinite ohms</td>
<td>Zero ohms</td>
</tr>
<tr>
<td>N.C.</td>
<td>Zero ohms</td>
<td>Infinite ohms</td>
</tr>
</tbody>
</table>

**Mastering Examine On & Examine Off Instructions**

- PLC programs have both normally open and normally closed input instructions.

![Diagram showing normally open and normally closed input instructions]
Mastering Examine On & Examine Off Instructions

- The Examine On Instruction

This input instruction examines the specified bit for a logic 1. If the bit is a 1, the instruction is true, otherwise the instruction is false.

This is generally known as a normally open input instruction.

Note: This instruction must be assigned a bit address, not a word address.
Mastering Examine On & Examine Off Instructions

- The **Examine Off** Instruction

  ![Instruction Diagram]

  This input instruction examines the specified bit for a logic 0. If the bit is a 0, the instruction is true, otherwise the instruction is false.

  This is generally known as a **normally closed** input instruction.

---

Mastering Examine On & Examine Off Instructions

- Examine On instructions are also called:
  - **XIC**, *eXamine If Closed*

- Examine Off instructions are also called:
  - **XIO**, *eXamine If Open*

**Note:** The terms Examine on, Examine off, Examine if closed (XIC), and Examine if open (XIO) are unique to Allen-Bradley PLCs.
Mastering Examine On & Examine Off Instructions

- Examine On and Examine Off Instructions that are True are highlighted green in the PLC programming software:

![Examine On and Off Instructions Diagram]

- A program can Examine On (or Examine Off) real inputs, real outputs, internal storage bits, timer done bits, etc.

![Examine On and Off Instructions Diagram]
Input and output field devices are wired to PLC discrete Input/Output (I/O) modules. How the system functions depends on the program!

- The inputs could be programmed as two inputs ANDed together:

- Or, the inputs could be programmed as two inputs ORed together:

In either case, the wiring is the same! The PLC program logically connects the input devices to the output actuators through the PLC program!
### Mastering Examine On & Examine Off Instructions

#### Input field devices wired to the PLC

<table>
<thead>
<tr>
<th>Electrical Continuity</th>
<th>Input Image Table Bit</th>
<th>Logical Continuity</th>
</tr>
</thead>
<tbody>
<tr>
<td>+24v Start (N.O.)</td>
<td>0 or 1</td>
<td>True if the bit is a 1</td>
</tr>
<tr>
<td>Current flows if button is pressed</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Output field devices wired to the PLC

<table>
<thead>
<tr>
<th>Electrical Continuity</th>
<th>Output Image Table Bit</th>
<th>Logical Continuity</th>
</tr>
</thead>
<tbody>
<tr>
<td>120 VAC FAN</td>
<td>0 or 1</td>
<td>The output image table bit is set to a 1 when the rung is true or is reset to a 0 when the rung is false</td>
</tr>
<tr>
<td>Output turns on when the bit in the output image table is a 1 (voltage is applied across the output terminals and current flows in the output circuit)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Scanning Process

- Scan refers to the continuous and sequential process of:
  - Reading the PLC inputs
  - Executing the ladder program (rung-by-rung)
  - Updating the PLC outputs

The Scanning Process

- The scan sequence can be broken into two functional parts:
  - The **Program Scan**
    - Scan the ladder program
  - The **I/O Update Scan**
    - Write outputs, Read inputs
The Scanning Process

**The Program Scan:**
- For each rung executed, the PLC processor will:
  - Examine the status of the input image table bits,
  - Solve the ladder logic in order to determine logical continuity (is the rung true?),
  - Update the appropriate output image table bits, if necessary.

**Note:** The output will not actually be energized until the I/O update part of the scan.

---

The Scanning Process

**The I/O Update Scan:**
- Copy the output image table status to the ALL of the output terminals (discrete output circuits)
  - Power is applied to the output device if it’s output image table bit has been previously set to a 1.

- Copy the status of ALL of the input terminals to the input image table
  - If an input is active (i.e., there is electrical continuity), the corresponding bit in the input image table will be set to a 1.
The Scanning Process

Read inputs
All input Terminals → Input Image Table

Solve the ladder program
(update output image table as necessary)

Update outputs
All output Terminals ← Output Image Table

In a ladder program, a specific output address (e.g., O:013/02) should **NOT** be referenced on more than one rung!
- This is sometimes called “duplicate coils”
- Using duplicate coils will cause unpredictable operation and should be avoided
- When using duplicate coils “the last rung wins”
- See example on next slide
Duplicate Coil Example

**Problem:** Rungs 11 and 19 both reference the same output address:

![Diagram of ladder logic with duplicate coil example]

**Solution:** Edit the ladder program as follows:

![Diagram of updated ladder logic]

Problem corrected, this output is only used once in the entire program.

---

The Scanning Process

- The actual scan time is a function of:
  - The speed of the processor module
  - The length of the ladder program
  - The type of instructions executed
  - The actual ladder true/false conditions (e.g., jump instructions, subroutines, etc.)
The Scanning Process

- The actual scan time is calculated and stored in the PLCs memory
  - The PLC computes the scan time each time the END instruction is executed
  - Scan time data can be monitored via the PLC programming software (e.g., RSLogix 5)
  - Scan time data is addressable and can therefore be referenced in the PLC program

- Typical scan time data includes:
  - The maximum scan time
  - The last scan time

---

The Scanning Process

- Allen-Bradley PLCs generally have 3 modes of operation:
  - **Run Mode**
    - When placed in the RUN mode, the processor begins the scanning process as previously described
  - **Program Mode**
    - When placed in the PROGRAM mode, the processor stops scanning the ladder program and (typically) all the outputs are turned off
  - **Test Mode**
    - The TEST mode is identical to the RUN mode, except all outputs are disabled (held in their off state)
The Scanning Process

Most Allen-Bradley processors (controllers) have a 3-position keyswitch:

- **REM (Remote)**
  - In the remote mode, the PLC programming software (e.g., RSLogix5000) can be used to place the controller into the remote program mode or the remote run mode.

- **RUN**
  - When keyswitch is placed in the RUN mode, the controller is switched into the run mode. The PLC programming software cannot change the controllers mode.

- **PROG (Program)**
  - When keyswitch is placed in the PROG mode, the controller is switched into the program mode. The PLC programming software cannot change the controllers mode.