Applications of Programmable Logic Controllers

DG31 34
Purpose

- **Unit purpose:**
  - This Unit is designed to introduce candidates to Programmable Logic Controllers (PLCs) and enable them to understand how PLCs are applied to control industrial processes.
  - The Unit allows candidates to develop the necessary knowledge and skills to allow them to understand the basic construction and operation of PLCs.
  - The Unit also provides candidates with the opportunity to develop practical programming skills to enable them to apply a PLC to simulate control of a specified industrial process.
Learning Outcomes

1. Explain and classify PLC hardware.
2. Describe the operation of PLC software.
3. Solve an industrial related control problem by the safe application of PLC technology.
Assessment - 1

• The assessment for Outcomes 1 and 2 in this Unit should be combined together into one written assessment paper.

• This paper should be taken by candidates at one single assessment event that should last one hour and thirty minutes.

• This assessment should be conducted under controlled, supervised conditions.
Assessment - 2

• Outcome 3 should be assessed by an assignment in which candidates are asked to complete a series of tasks to enable them to apply a PLC to simulate the safe control of a specified industrial process.

• The assignment tasks should involve devising the control strategy, writing PLC software, programming of the PLC, verifying correct operation of the program and documentation of the final solution.
Assessment - 2

• Candidates should complete the assignment in eight hours.
• Candidates should have access to a PLC and be allowed to use any relevant course notes, textbooks and reference material for the PLC.
• Both the written assessment and the practical assignment should be carried out at the end of the delivery of the Unit.
Programmable Logic Controller
What is a PLC

A Programmable Logic Controller (PLC) is a mini computer specifically designed for industrial use.
PLC Origin

Developed to replace relays in the late 1960s

Costs dropped and became popular by 1980s

Now used in many industrial designs
Purpose of PLC

Examples of PLC Applications are:

- Pneumatic and Hydraulic Machines
- Robots
- Production Processes
- Packaging Lines
- Signalling Systems (Traffic Lights etc)
- Refining Processes
Purpose of PLC

• Old control room
Purpose of PLC

- Modern control room
Architecture of PLC

A/D CONVERTER  COMMUNICATION INTERFACE  PROGRAMMER

INPUT MODULE  ROM  OUTPUT MODULE  RAM  CPU  MONITOR
Architecture of PLC
Classification of PLC’s

PLCs are Classified by :

• Number of Input / Output terminals
• Available Memory
• Into three categories
  • Large
  • Medium
  • Small
Classification of PLC’s

• Classifying any microprocessor system in terms of memory size tends to be out of date the moment it is printed.

• As a result use the following as a guide only.

<table>
<thead>
<tr>
<th>PLC Classification</th>
<th>I/O Count</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large PLC</td>
<td>&gt; 1024</td>
<td>&gt; 10kB RAM</td>
</tr>
<tr>
<td>Medium PLC</td>
<td>&lt; 1024</td>
<td>≤ 10kB RAM</td>
</tr>
<tr>
<td>Small PLC</td>
<td>&lt; 256</td>
<td>&gt;1kB RAM</td>
</tr>
<tr>
<td>Micro PLC</td>
<td>≤ 32</td>
<td>&lt; 1kB RAM</td>
</tr>
<tr>
<td>Nano PLC</td>
<td>≤ 16</td>
<td>&lt; 1kB RAM</td>
</tr>
</tbody>
</table>

Typical Classification of PLC by the Number of I/O

• Overlaps are likely due to available expansion modules.
Types of PLC

Unitary

The Unitary PLC contains every feature of a basic system in one box. They are attached to the machine being controlled.

Modular

Modular use a range of modules that slot together to build up a system. The basic modules are the power supply, the main module containing the CPU, the input module and the output module. Other modules such as A/D converters may be added. The main advantage is that the number of input and output terminals can be expanded to cope with changes to the hardware system.

Rack Mounted

This is a similar concept to the modular design but the modules are on standard cards that slot into a standard rack inside a cabinet. These are flexible and allow expansion of the system.
Inputs (Sensors)

Position (linear and angular)
- Temperature
- Speed
- Pressure
- Weight
- Quantity
- Flow rate
- Depth
- Density
- Acidity
- Voltage
- Current
- Torque
- Power

- Proximity Switches
- Switches
- Potentiometer
- LVDTs
Input Module Protection

+24V

PLC Input

0V

Status LED

Filter

Live

AC input

Neut

optocoupler

+5V

TTL
Outputs (Actuators)
Conveyor Belt
Linear Electric Actuator
Steam & High Pressure Solenoid Valves

High Pressure Steam Solenoid

Water Solenoid Valve
Traffic Lights
Alarms
Heating Elements
PLC Status

There are usually a number of visible status indicators on a PLC.

For example:
- Power On
- Program Running
- Programming Mode
- Fault

They do not all necessarily appear on every PLC.
Communicating with a PLC

- Computer
- PLC
- Store B
- Store A
- Make B
- Make A
- Assembly
- Inspection
- Packaging
- Warehouse

Cables:
- COAXIAL CABLES
- TWISTED PAIR
- FIBRE OPTICS
- RIBBON CABLES
The Scan Cycle

1. HOUSE KEEPING
   - PLC Self test

2. INPUT MODULE
   - Check I/P status

3. PROGRAM LOGIC
   - Apply current logic

4. OUTPUT MODULE
   - Set/reset O/P status
Scan Cycle

PLC program changes outputs by examining inputs

Set new outputs

The Control Loop

Power turned on

Read inputs

Process changes and PLC pauses while it checks its own operation
Response Time

**SELF TEST** - Checks to see if all cards error free, reset watch-dog timer, etc. *(A watchdog timer will cause an error, and shut down the PLC if not reset within a short period of time - this would indicate that the ladder logic is not being scanned normally).*

**INPUT SCAN** - Reads input values from the chips in the input cards, and copies their values to memory. This makes the PLC operation faster, and avoids cases where an input changes from the start to the end of the program (e.g., an emergency stop). There are special PLC functions that read the inputs directly, and avoid the input tables.

**PROGRAM LOGIC SOLVE/SCAN** - Based on the input table in memory, the program is executed 1 step at a time, and outputs are updated. This is the focus of the later sections.

**OUTPUT SCAN** - The output table is copied from memory to the output chips. These chips then drive the output devices.
Any Questions
PLC Faults

The PLC has certain diagnostic, monitoring and testing facilities within the software.

On most PLC’s Light Emitting Diodes (LED’s) shows the status of the inputs and outputs, some show status by way of a Liquid Crystal Display (LCD).

It is also possible to fix a bank of switches to the input side and test a programme by setting the switches to a known states and seeing if the appropriate output action is taken.

The most advanced method - connects the PLC to a computer with appropriate software and runs a complete simulation of the system being controlled showing the status of everything.
## Fault Finding Guide

### PLC Faults

<table>
<thead>
<tr>
<th>Symptom</th>
<th>Likely cause</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total failure or stoppage of the automation system</td>
<td>Power failure &lt;br&gt;Supply turned off &lt;br&gt;Emergency stop device activated</td>
<td>Every area of the plant will be affected &lt;br&gt;if power lamp is off, check the main switch or circuit-breaker &lt;br&gt;Inspect each emergency device and reset if it safe to do so</td>
</tr>
<tr>
<td>All inputs fail</td>
<td>Short circuit or earth fault on sensor or wiring</td>
<td>Consider stopping program. Disconnect inputs one by one until the input voltage returns</td>
</tr>
<tr>
<td>All outputs fail</td>
<td>Control circuit fuse or <strong>miniature circuit-breaker (mcb) operated</strong> &lt;br&gt;Program stopped</td>
<td>Test the control circuit supply, using a voltmeter. Repair or reset as necessary &lt;br&gt;Run lamp not on: CPU/MEM lamp may indicate an error. Secure safety functions and restart program</td>
</tr>
<tr>
<td>Group of outputs fail</td>
<td>Fuse or (mcb) for that group of outputs has operated &lt;br&gt;Faulty output module</td>
<td>Output LEDs on: short circuit or earth fault may be the cause; check for voltage and repair or reset as necessary &lt;br&gt;Output LEDs are off: replace module using recommended procedure</td>
</tr>
<tr>
<td>Single output device fails to turn on</td>
<td>Wiring fault &lt;br&gt;Device fault &lt;br&gt;Absent input signal</td>
<td>Output LED is on. voltage at output is normal and voltage at device is absent &lt;br&gt;Output LED is on. voltage at output is normal and voltage at device is normal &lt;br&gt;Find out which one by reading ladder and checking LEDs or by monitoring the program</td>
</tr>
<tr>
<td>Single output device fails to turn on</td>
<td>Module fault</td>
<td>Confirm that all signals for turning on the output are correct. Check output LED and output voltage. Replace module, etc.</td>
</tr>
<tr>
<td>Output fails to turn off</td>
<td>Control signals are keeping it on &lt;br&gt;Module switching device</td>
<td>Find out which one by reading ladder and checking LEDs or by monitoring the program &lt;br&gt;Output LED is off but the destroyed output itself remains on relay contacts welded, transistor or triac faulty) Replace module</td>
</tr>
<tr>
<td>Program stopped</td>
<td>Mode set to <strong>STOP</strong> by hand or by program &lt;br&gt;CPU or MEM fault</td>
<td>Set to <strong>RUN</strong> if safe to do so &lt;br&gt;Fault lamp confirms this. Reload good copy of program and set to run. Use diagnostic functions to check scan time; if max. value has been exceeded analyze program to find a means of reducing it</td>
</tr>
<tr>
<td><strong>No</strong> program</td>
<td>Loss of power to RAM memory</td>
<td>Loss battery lamp is on. Replace batter, reload program</td>
</tr>
<tr>
<td>Intermittent fault or spurious operation</td>
<td>Gross electromagnetic interference or lightning damage</td>
<td>Shield controller from possible sources of interference: check over-voltage protection and reload program</td>
</tr>
</tbody>
</table>