Ch 9  Discrete Control
Using PLCs and PCs

Sections:
1. Discrete Process Control
2. Ladder Logic Diagrams
3. Programmable Logic Controllers
4. Personal Computers Using Soft Logic
Discrete Process Control

Control systems that operate on parameters and variables that change at discrete moments in time

- Parameters and variables are also discrete, usually binary (0 or 1, off or on, open or closed, etc.)

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Limit switch</th>
<th>Contact/no contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Photo-detector</td>
<td>On/off</td>
<td></td>
</tr>
<tr>
<td>Timer</td>
<td>On/off</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actuators</th>
<th>Motor</th>
<th>On/off</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valve</td>
<td>Open/closed</td>
<td></td>
</tr>
<tr>
<td>Clutch</td>
<td>Engaged/not engaged</td>
<td></td>
</tr>
</tbody>
</table>
Categories of Discrete Control

1. Logic control – event-driven changes
2. Sequencing – time-driven changes
Logic Control

A switching system whose output at any moment is determined exclusively by the values of inputs

- No memory
- No operating characteristics that depend on time
- Also called *combinational logic control*
Elements of Logic Control

- Basic elements, called logic gates:
  - AND – output = 1 if all inputs = 1, zero otherwise
  - OR – output = 1 if any input = 1, zero otherwise
  - NOT – output = 1 if (single) input = 0, and vice versa

- Additional elements:
  - NAND – combination of AND and NOT
  - NOR – combination of OR and NOT
AND Gate

Electrical circuit illustrating the operation of the logical AND gate.
OR Gate

Electrical circuit illustrating the operation of the logical OR gate.
NOT Gate

Electrical circuit illustrating the operation of the logical NOT gate.
## Symbols for Logical Gates: U.S. and ISO

<table>
<thead>
<tr>
<th></th>
<th>U.S. symbol</th>
<th>ISO symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND</td>
<td><img src="image" alt="AND U.S. symbol" /></td>
<td><img src="image" alt="AND ISO symbol" /></td>
</tr>
<tr>
<td>OR</td>
<td><img src="image" alt="OR U.S. symbol" /></td>
<td><img src="image" alt="OR ISO symbol" /></td>
</tr>
<tr>
<td>NOT</td>
<td><img src="image" alt="NOT U.S. symbol" /></td>
<td><img src="image" alt="NOT ISO symbol" /></td>
</tr>
<tr>
<td>NAND</td>
<td><img src="image" alt="NAND U.S. symbol" /></td>
<td><img src="image" alt="NAND ISO symbol" /></td>
</tr>
<tr>
<td>NOR</td>
<td><img src="image" alt="NOR U.S. symbol" /></td>
<td><img src="image" alt="NOR ISO symbol" /></td>
</tr>
</tbody>
</table>
Sequencing

A switching system that uses internal timing devices to determine when to initiate changes in output variables

- Outputs are usually generated “open loop”
  - No feedback that control function is executed
- Sequence of output signals is usually cyclical, as in a high production work cycle
  - The signals occur in the same repeated pattern within each regular cycle
- Common sequencing devices:
  - Timer – output switches on/off at preset times
  - Counter – counts electrical pulses and stores them
Ladder Logic Diagrams

A diagram in which various logic elements and other components are displayed along horizontal rungs connected on either end to two vertical rails

- Types of elements and components:
  1. Contacts - logical inputs (usually), e.g., limit switches, photo-detector
  2. Loads - outputs, e.g., motors, lights, alarms, solenoids
  3. Timers - to specify length of delay
  4. Counters - to count pulses received
## Symbols for Common Elements Used in Ladder Logic Diagrams

<table>
<thead>
<tr>
<th>Ladder symbol</th>
<th>Hardware component</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>Normally open contacts (switch, relay, other ON/OFF devices)</td>
</tr>
<tr>
<td>(b)</td>
<td>Normally closed contacts (switch, relay, etc.)</td>
</tr>
<tr>
<td>(c)</td>
<td>Output loads (motor, lamp, solenoid, alarm, etc.)</td>
</tr>
<tr>
<td>(d) [TMR 3s]</td>
<td>Timer</td>
</tr>
<tr>
<td>(e) [CTR]</td>
<td>Counter</td>
</tr>
</tbody>
</table>
Advantages of Ladder Logic Diagrams

- Familiar to shop personnel who must construct, test, maintain, and repair the control system
- Analogous to the electrical circuits used to accomplish logic and sequence control
- Principal technique for programming PLCs
Programmable Logic Controller (PLC)

A microcomputer-based controller that uses stored instructions in programmable memory to implement logic, sequencing, timing, counting, and arithmetic functions through digital or analog modules, for controlling machines and processes

- Applications in both process industries and discrete manufacturing
- Introduced around 1970
- Replaced hard-wired electromechanical relay panels
Components of a PLC

External source of power

Power supply

Programming device

Processor

Input/output module

Memory unit

Inputs

Outputs
Advantages of PLCs Compared to Relay Control Panels

- Programming a PLC is easier than wiring a relay control panel
- PLC can be reprogrammed
- PLCs take less floor space
- Greater reliability, easier maintenance
- PLC can be connected to computer systems (CIM)
- PLCs can perform a greater variety of control functions
PLC Components

1. Processor – executes logic and sequencing functions by operating on the PLC inputs to determine the appropriate output signals
2. Input/output module – connections to process
3. Memory unit – contains the programs of logic, sequencing, and I/O operations
4. Power supply – converts 120 V (ac) to dc voltages of $\pm$5 V compatible with process equipment
5. Programming device
Typical PLC Operating Cycle

1. Input scan – inputs are read by processor and stored in memory
2. Program scan – control program is executed
   - Input values stored in memory are used in the control logic calculations to determine values of outputs
3. Output scan – output values are updated to agree with calculated values
   - Time to perform the three steps (scan time) varies between 1 and 25 msec
Additional PLC Capabilities

- Analog control – PID control available on some PLCs for continuous processes
- Arithmetic functions – permits more complex control algorithms to be implemented than conventional logic and sequencing elements
- Matrix functions – e.g., linear programming for optimal control
- Data processing and reporting – business applications
  - Blurs the distinction between PLCs and PCs
PLC Programming

- **Graphical languages:**
  1. Ladder logic diagrams – most widely used
  2. Function block diagrams – instructions composed of operation blocks that transform input signals
  3. Sequential function charts – series of steps and transitions from one state to the next (Europe)

- **Text-based languages:**
  1. Instruction list - low-level computer language
  2. Structured text – high-level computer language
Personal Computers Using Soft Logic

- Available in sturdy enclosures for plant environment
- Membrane-type keyboards to protect against dirt, moisture, etc.
- Can be ordered with I/O cards and other hardware to connect to machines and processes
- Installed with Windows for implementing control applications
- Can be programmed with soft logic - software that emulates the operations of the built-in control software in PLCs