

Introduction to Problem Solving in Industrial Automation and Robotics

Introduction to Problem Solving in Industrial Automation and Robotics
(A concise, easy-to-read study guide)

1. What Is Problem Solving in This Context?

Aspect	Definition
Industrial Automation	Use of machines, control systems, and information technologies to perform manufacturing tasks with minimal human intervention.
Robotics	Design, construction, operation, and use of robots to carry out tasks that are repetitive, hazardous, or require high precision.
Problem Solving	Systematic approach to identify, analyze, and resolve issues that prevent a process, machine, or robot from meeting its performance goals.

Goal: Deliver reliable, efficient, and safe production while minimizing downtime and cost.

2. Core Problem Solving Frameworks

Framework	Steps	Typical Use
IDEAL (Identify ? Define ? Explore ? Act ? Look back)	1. Spot the symptom 2. Clarify the problem statement 3. Generate alternatives 4. Implement the best solution 5. Review results	General troubleshooting, process improvement
DMAIC (Define, Measure, Analyze, Improve, Control)	1. Define project scope 2. Measure current performance 3. Analyze root causes 4. Improve with pilot changes 5. Control to sustain gains	Six Sigma projects, quality-focused upgrades
APM (Analyze?Plan?Monitor)	1. Analyze data & failure modes 2. Plan corrective actions 3. Monitor implementation & results	Real-time fault detection, predictive maintenance

Tip: Choose the framework that matches the problem's complexity and the organization's culture.

3. Typical Problem Types in Automation & Robotics

Introduction to Problem Solving in Industrial Automation and Robotics

Category	Examples	Common Indicators
Mechanical	Misaligned drives, loose belts, worn bearings, joint backlash	Vibration, unusual noises, reduced torque
Electrical / Power	Faulty motor drives, supply voltage sag, grounding issues	Tripped breakers, erratic motor speeds
Control / Software	PLC logic errors, communication timeouts, sensor drift	Unexpected PLC states, alarms, data outliers
Integration	Mismatched data formats, timing conflicts between subsystems	Synchronization errors, missed cycles
Safety	Emergency stop not responding, safety light curtain failure	Safety interlock trips, audit non-compliance

4. Structured Troubleshooting Process

1. Gather Information

- Review operator logs, HMI screens, and alarm histories.
- Capture real-time data (current, voltage, position, speed).

2. Isolate the Symptom

- Use “divide and conquer”: test subsystems independently (e.g., motor ? drive ? controller).

3. Form Hypotheses

- Apply **Cause-Effect (Fishbone) Diagrams** or **Fault Tree Analysis** to list possible root causes.

4. Test Hypotheses

- Perform controlled experiments (swap components, simulate inputs).
- Record results in a **troubleshooting log**.

5. Implement Corrective Action

- Apply the fix (replace part, update firmware, adjust PID).

Introduction to Problem Solving in Industrial

- Follow change management procedures (approval, documentation).

Automation and Robotics

6. Validate & Document

- Verify that the system meets performance specs.
 - Update **Standard Operating Procedures (SOPs)** and **maintenance manuals**.
-

5. Key Tools & Techniques

Tool Purpose Typical Application